53-1002720-02
25 Month 2013

Fabric OS Encryption

Administrator's Guide Supporting RSA Data Protection Manager (DPM) Environments

Supporting Fabric OS v7.1.0

BROCADE

Notice: This document is for informational purposes only and does not set forth any warranty, expressed or implied, concerning any equipment, equipment feature, or service offered or to be offered by Brocade. Brocade reserves the right to make changes to this document at any time, without notice, and assumes no responsibility for its use. This informational document describes features that may not be currently available. Contact a Brocade sales office for information on feature and product availability. Export of technical data contained in this document may require an export license from the United States government.

The authors and Brocade Communications Systems, Inc. shall have no liability or responsibility to any person or entity with respect to any loss, cost, liability, or damages arising from the information contained in this book or the computer programs that accompany it.

The product described by this document may contain “open source” software covered by the GNU General Public License or other open source license agreements. To find out which open source software is included in Brocade products, view the licensing terms applicable to the open source software, and obtain a copy of the programming source code, please visit *http://www.brocade.com/support/oscd*.

Brocade Communications Systems, Incorporated

Corporate and Latin American Headquarters
Brocade Communications Systems, Inc.
130 Holger Way
San Jose, CA 95134
Tel: 1-408-333-8000
Fax: 1-408-333-8101
E-mail: info@brocade.com

Asia-Pacific Headquarters
Brocade Communications Systems China HK, Ltd.
No. 1 Guanghua Road
Chao Yang District
Units 2718 and 2818
Beijing 100020, China
Tel: +8610 6588 8888
Fax: +8610 6588 9999
E-mail: china-info@brocade.com

European Headquarters
Brocade Communications Switzerland Sàrl
Centre Swissair
Tour B - 4ème étage
29, Route de l'Aéroport
Case Postale 105
CH-1215 Genève 15
Switzerland
Tel: +41 22 799 5640
Fax: +41 22 799 5641
E-mail: emea-info@brocade.com

Asia-Pacific Headquarters
Brocade Communications Systems Co., Ltd. (Shenzhen WFOE)
Citic Plaza
No. 233 Tian He Road North
Unit 1308 – 13th Floor
Guangzhou, China
Tel: +8620 3891 2000
Fax: +8620 3891 2111
E-mail: china-info@brocade.com

Document History

Title	Publication number	Summary of changes	Date
Fabric OS Encryption Administrator’s Guide for RSA Data Protection Manager (DPM) Environments	*53-1002720-02*	Added HA cluster information and reviewer comments	March 2013
Fabric OS Encryption Administrator’s Guide for RSA Data Protection Manager (DPM) Environments	*53-1002720-01*	New document	December 2012

Contents

About This Document

Chapter 1 Encryption Overview

Chapter 2 Configuring Encryption Using the Management Application

Chapter 3 **Configuring Encryption Using the CLI**

chapter 4 **Deployment Scenarios**

Chapter 5 **Best Practices and Special Topics**

Chapter 6 Maintenance and Troubleshooting

Appendix A State and Status Information

Index

About This Document

In this chapter

How this document is organized

. This document is organized to help you find the information that you want as quickly and easily as possible.

The document contains the following components:

- Chapter 1, "Encryption Overview," provides a task matrix, an overview of the data encryption switch and the encryption solution, and the terminology used in this document.
- Chapter 2, "Configuring Encryption Using the Management Application," describes how to configure and manage encryption features using Brocade Network Advisor.
- Chapter 3, "Configuring Encryption Using the CLI," describes how to configure and manage encryption features using the command line interface.
- Chapter 4, "Deployment Scenarios," describes SAN configurations in which encryption may be deployed.
- Chapter 5, "Best Practices and Special Topics," summarizes best practices and addresses special topics relevant to the implementation of encryption features.
- Chapter 6, "Maintenance and Troubleshooting," provides information on troubleshooting and the most common commands and procedures to use to diagnose and recover from problems.
- Appendix A, "State and Status Information," lists the encryption engine security processor (SP) states, security processor key encryption key (KEK) status information, and encrypted LUN states.

Supported hardware and software

The following hardware platforms support data encryption as described in this manual.

- Brocade DCX Backbone series chassis with an FS8-18 encryption blade.
- Brocade Encryption Switch.
- If you are upgrading your Fabric OS installation to v7.1.0, you must first update your key management server from RKM 2.x to DPM 3.x. (DPM 3.2.1 is currently supported with Fabric OS 7.1.0.)

What's new in this document

This document identifies any encryption changes that support Fabric OS 7.1.0.

Document conventions

This section describes text formatting conventions and important notice formats used in this document.

Text formatting

The narrative-text formatting conventions that are used are as follows:

bold text	Identifies command names Identifies the names of user-manipulated GUI elements Identifies keywords and operands Identifies text to enter at the GUI or CLI
italic text	Provides emphasis Identifies variables Identifies paths and Internet addresses Identifies document titles
`code` text	Identifies CLI output Identifies command syntax examples

For readability, command names in the narrative portions of this guide are presented in mixed lettercase: for example, switchShow. In actual examples, command lettercase is often all lowercase. Otherwise, this manual specifically notes those cases in which a command is case sensitive.

Command syntax conventions

Command syntax in this manual follows these conventions:

command	Commands are printed in bold.
--**option, option**	Command options are printed in bold.
-**argument,** arg	Arguments.
[ ]	Optional element.
variable	Variables are printed in italics. In the help pages, variables are underlined or enclosed in angled brackets < >.
...	Repeat the previous element, for example "member[;member...]"
value	Fixed values following arguments are printed in plain font. For example, --**show** WWN
|	Boolean. Elements are exclusive. Example: --**show** -**mode** egress | ingress
\	Backslash. Indicates that the line continues through the line break. For command line input, type the entire line without the backslash.

Notes, cautions, and warnings

The following notices and statements are used in this manual. They are listed below in order of increasing severity of potential hazards.

NOTE
A note provides a tip, guidance or advice, emphasizes important information, or provides a reference to related information.

ATTENTION
An Attention statement indicates potential damage to hardware or data.

CAUTION

A Caution statement alerts you to situations that can cause damage to hardware, firmware, software, or data.

DANGER

A Danger statement indicates conditions or situations that can be potentially lethal or extremely hazardous to you. Safety labels are also attached directly to products to warn of these conditions or situations.

Key terms

For definitions specific to Brocade and Fibre Channel, see the technical glossaries on Brocade Connect. See “Brocade resources” on page xvi for instructions on accessing MyBrocade.

For definitions specific to this document, see “Terminology” on page 2.

For definitions of SAN-specific terms, visit the Storage Networking Industry Association online dictionary at:

http://www.snia.org/education/dictionary

Additional information

This section lists additional Brocade and industry-specific documentation that you might find helpful.

Brocade resources

To get up-to-the-minute information, go to http://my.brocade.com and register at no cost for a user ID and password.

For practical discussions about SAN design, implementation, and maintenance, you can obtain *Building SANs with Brocade Fabric Switches* through:

http://www.amazon.com

For additional Brocade documentation, visit the Brocade SAN Info Center and click the Resource Library location:

http://www.brocade.com

Release notes are available on the MyBrocade website and are also bundled with the Fabric OS firmware.

Other industry resources

- White papers, online demos, and data sheets are available through the Brocade website at *http://www.brocade.com/products-solutions/products/index.page*.
- Best practice guides, white papers, data sheets, and other documentation is available through the Brocade Partner website.

For additional resource information, visit the Technical Committee T11 Web site. This website provides interface standards for high-performance and mass storage applications for Fibre Channel, storage management, and other applications:

http://www.t11.org

For information about the Fibre Channel industry, visit the Fibre Channel Industry Association website:

http://www.fibrechannel.org

For information about the Key Management Interoperability Protocol standard, visit the OASIS KMIP Technical Committee website:

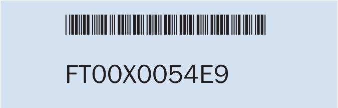
https://www.oasis-open.org/committees/tc_home.php?wg_abbrev=kmip

Getting technical help

Contact your switch support supplier for hardware, firmware, and software support, including product repairs and part ordering. To expedite your call, have the following information available:

1. General Information
   - Switch model
   - Switch operating system version
   - Error numbers and messages received
   - **supportSave** command output
   - Detailed description of the problem, including the switch or fabric behavior immediately following the problem, and specific questions
   - Description of any troubleshooting steps already performed and the results
   - Serial console and Telnet session logs
   - syslog message logs
2. Switch Serial Number

   The switch serial number and corresponding bar code are provided on the serial number label, as illustrated below.:



   The serial number label is located as follows:

   - *Brocade Encryption Switch*—On the switch ID pull-out tab located inside the chassis on the port side of the switch on the left.
   - *Brocade DCX*—On the bottom right on the port side of the chassis
   - *Brocade DCX-4S*—On the bottom right on the port side of the chassis, directly above the cable management comb.
   - *Brocade DCX 8510-8*—On the port side of the chassis, on the lower right side, and directly above the cable management comb.
   - *Brocade DCX 8510-4*—On the port side of the chassis, on the lower right side and directly above the cable management comb.
3. World Wide Name (WWN)

   Use the **licenseIdShow** command to display the WWN of the chassis.

If you cannot use the **licenseIdShow** command because the switch is inoperable, you can get the WWN from the same place as the serial number, except for the Brocade DCX. For the Brocade DCX, access the numbers on the WWN cards by removing the Brocade logo plate at the top of the non-port side of the chassis.

Document feedback

Quality is our first concern at Brocade and we have made every effort to ensure the accuracy and completeness of this document. However, if you find an error or an omission, or you think that a topic needs further development, we want to hear from you. Forward your feedback to:

documentation@brocade.com

Provide the title and version number of the document and as much detail as possible about your comment, including the topic heading and page number and your suggestions for improvement.

Chapter

Encryption Overview

1

In this chapter

Host and LUN considerations

Encrypting data-at-rest provides peace of mind in terms of protecting data from loss or theft, but very careful planning must be done to ensure encrypted data is handled correctly. Much of the planning must come from careful evaluation of host application and LUN resources, and of the path that the data will take to get from one or more hosts to a LUN.

CAUTION

When implementing encryption for data-at-rest, all hosts that access a LUN that is to hold encrypted data need to be configured for encryption to avoid data corruption. If a host, possibly in another fabric, writes cleartext to an encrypted LUN, the data on the LUN will be lost. The user must ensure that all hosts that can access a LUN are configured in the same manner.

Terminology

The following are definitions of terms used extensively in this document.

ciphertext Encrypted data.

cleartext Unencrypted data.

CryptoModule The secure part of an encryption engine that is protected to the FIPS 140-2 level 3 standard. The term CryptoModule is used primarily in the context of FIPS authentication.

Data Encryption Key (DEK) An encryption key generated by the encryption engine. The DEK is used to encrypt cleartext received from a host before it is sent to a target LUN, and to decrypt that data when it is retrieved by the host.

Data Encryption Key Cluster (DEK Cluster) A cluster of encryption engines which can host all paths to a LUN and share the same data encryption key (DEK) set. The encryption engines can be in the same or different fabrics. DEK clusters enable host MPIO failover.

Encryption Engine The entity within a node that performs encryption operations, including the generation of Data Encryption Keys.

Encryption Group A collection of one or more DEK clusters, HA clusters, or both, which share the same key vault and device configuration, and is managed as a single group.

Failback In the context of this implementation of encryption, failback refers to behavior after a failed encryption switch recovers. Devices that were transferred to another switch by failover processing may automatically be transferred back, or they may be manually switched back. This is determined as a configuration option.

Failover In the context of this implementation of encryption, failover refers to the automatic transfer of devices hosted by one encryption switch to another encryption switch within a high availability cluster (HA cluster).

Group Leader A group leader is a special node within an encryption group which acts as a group and cluster manager, and manages and distributes all group-wide and cluster-wide configurations to all members of the group or cluster.

High Availability Cluster (HA Cluster) A collection of peer-level encryption engines that provide failover capabilities within a fabric.

Key Encryption Key A key used to encrypt and decrypt Data Encryption Keys (DEKs) within encryption devices so that DEKs are transmitted in a secure manner outside of the encryption engines, and stored persistently inside key vaults.

Link Key A shared secret exchanged between an encryption engine and a FIPS 140-2 level 3 certified key management appliance and key vault. The link key is an Key Encryption Key (KEK) that is used to encrypt Data Encryption Keys (DEKs) in transit over a secure connection to and from the key vault. The key management appliance decrypts the DEKs and stores them encrypted with its own master key.

Logical Unit Number (LUN) The identifier of a SCSI logical unit.

Master Key A Key Encryption Key (KEK) used to encrypt and decrypt DEKs when storing DEKs in opaque key vaults. There is one master key per encryption group. That means all node encryption engines within an encryption group use the same master key to encrypt and decrypt the DEKs.

Node In terms of encryption, a Brocade Encryption Switch, DCX, or DCX-4S through which users can manage an encryption engine.

Opaque Key Vault A storage location that provides untrusted key management functionality. Its contents may be visible to a third party. DEKs in an opaque key vault are stored encrypted in a master key to protect them.

Recovery cards A set of smart cards that contain a backup master key. Each recovery card holds a portion of the master key. The cards must be gathered and read together from a card reader attached to a PC running the BNA client to restore the master key. Recovery cards may be stored in different locations, making it very difficult to steal the master key. The cards should not be stored together, as that defeats the purpose.

Redirection zone When encryption is implemented, data traffic is routed to and from virtual initiators and virtual targets. Redirection zones are automatically created to enable frame redirection to the virtual initiators and virtual targets.

Rekeying Rekeying refers to decrypting data with the current Data Encryption Key (DEK), and encrypting it with a new DEK. This is done when the security of the current key is compromised, or when a DEK is configured to expire in a specific time frame. The rekeying operation can be used to encrypt existing data currently stored as cleartext. In that case, there is no existing DEK, and the data does not have to be decrypted before it is encrypted using the new DEK.

Trusted Key Vault Very secure storage on a hardware appliance that establishes a trusted link with the encryption device for secure exchange of DEKs. DEKs are encrypted with the link for transit between the encryption device and the hardware appliance. At the hardware appliance, the DEKs are re-encrypted, using master key created and maintained by hardware appliance, and then stored in the trusted key vault.

Virtual Initiator A logical entity that acts as a stand-in for a physical host when communicating with a physical target LUN.

Virtual Target A logical entity that acts as a stand-in for a physical target LUN when communicating with a physical host. A virtual target is mapped one to one to a specific physical target.

The Brocade Encryption Switch

The Brocade Encryption Switch is a high-performance, 32-port, auto-sensing 8 Gbps Fibre Channel switch with data cryptographic (encryption/decryption) and data compression capabilities. The switch is a network-based solution that secures data-at-rest for heterogeneous tape drives, disk array LUNs, and virtual tape libraries by encrypting the data using Advanced Encryption Standard (AES) 256-bit algorithms. Encryption and decryption engines provide in-line encryption services with up to 96 Gbps throughput for disk I/O (mix of ciphertext and cleartext traffic) and up to 48 Gbps throughput for tape I/O (mix of ciphertext and cleartext traffic). Refer to "The FS8-18 blade" on page 5 for information about license requirements for 48 Gbps and 96 Gbps throughput.

In addition to its 32 Fibre Channel ports, the switch has one RJ45 Gigabit Ethernet (GE) management port, two RJ45 GE ports for clustering interconnection and rekey synchronization, one RJ-45 Serial console port, and one USB port for serviceability, error logging, and firmware upgrades (Figure 1) .

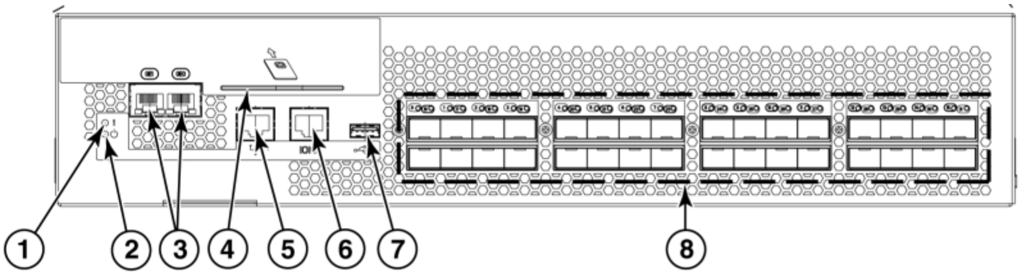


FIGURE 1 **Brocade Encryption Switch**

1 Power LED.

2 Status LED.

3 RJ-45 gigabit Ethernet ports (labeled eth0 and eth1) for clustering and centralized management of multiple encryption switches through a group leader.

4 Smart card reader.

5 RJ-45 gigabit Ethernet port for the management interface. This interface is used for the secure connection to the key vault location and to the BNA client.

6 RJ-45 serial console port.

7 USB port for firmware upgrades and other support services.

8 Fibre Channel ports (0-31) - 1, 2, 4, or 8 Gbps auto-sensing F, FL, E, EX, or M ports to connect host servers, SAN disks, SAN tapes, edge switches, or core switches.

The FS8-18 blade

The FS8-18 blade provides the same features and functionality as the Brocade Encryption Switch. The FS8-18 blade installs on the Brocade DCX Backbone chassis, which include the DCX, DCX-4S, DCX 8510-8, and DCX 8510-4 chassis.

FIPS mode

Both the Brocade Encryption Switch and the FS8-18 blade always boot up in FIPS mode, which cannot be disabled. In this mode, only FIPS-compliant algorithms are allowed.

Performance licensing

Encryption processing power is scalable, and may be increased by purchasing and installing an encryption performance license. The base unit Brocade Encryption Switch and FS8-18 Encryption Blade have a standard capacity of 48 Gbps of encryption processing power. Additional encryption processing power can be added for disk I/O by purchasing and installing an Advanced Disk Encryption Performance Upgrade license. When the performance upgrade license is applied, encryption processing power of up to 96 Gbps is available for disk encryption. Note that when the license is applied to a Brocade DCX Backbone chassis, it applies to all FS8-18 blades installed on that chassis.

Adding a license

The encryption performance licenses are added just like any other Fabric OS feature license. After the license is added, the Brocade Encryption Switch and Brocade DCX Backbone chassis with encryption blades installed must be rebooted for the license to take effect. See the *Fabric OS Administrator's Guide* for information about obtaining and adding licenses.

Licensing best practices

Licenses installed on the switches and blades must have identical performance numbers when used together in high availability (HA) clusters or data encryption key (DEK) clusters.

Recommendation for connectivity

In order to achieve high performance and throughput, the encryption engines perform what is referred to as "cut-through" encryption. In simple terms, this is achieved by encrypting the data in data frames on a per-frame basis. This enables the encryption engine to buffer only one frame, encrypt it, and send out the frame to the target on write I/Os. For read I/Os, the reverse is done.

This puts some constraints on the topology and the container configurations to support acceptable performance for encrypted and decrypted I/O to and from LUNs, and to support acceptable levels of scale in terms of the number of LUNs and the number of flows. The topology and container configuration constraint are stated below:

Care must be taken when connecting the encryption engines to the fabric and configuring crypto-target containers to be sure that the traffic flow between the host initiator and the physical storage array LUN through the container flows through only one encryption engine that is hosting the container. This is to avoid crisscrossing of flows to and from virtual entities; that is, from virtual targets and virtual initiators on two different encryption engines over the same path.

Although there is considerable flexibility in connecting and configuring the containers for encryption, the following guidelines are the recommended best practices:

- Host and storage array ports that are not involved in any encryption flow can be connected to any encryption engines (EEs).
- Recommendations for host and target ports with respect to encryption flows are as follows:
  - For high availability (HA) purposes, only ISLs are connected to the encryption engine to connect it to the fabric. No devices (initiators and targets) are connected to it.
  - To maintain HA, we recommend that devices (hosts and targets) and ISLs not be connected directly to the encryption blades (FS8-18) in a Brocade DCX Backbone chassis in a single-path configuration.

Usage limitations

There are usage limitations to be aware of when planning an encryption implementation:

- Special redirection zones are created to handle data that is redirected to an encryption switch or blade. Quality of Service (QoS) cannot be applied to a redirection zone.
- For frame redirection to be applied, regular zones for hosts and targets must be defined in the effective configuration. Hosts and targets must be zoned together by worldwide port name (WWPN) rather than worldwide node name (WWNN) in configurations where frame redirection will be used. If hosts or targets are zoned together using worldwide node name, frame redirection will not occur properly.

NOTE
The use of alias names in place of WWPNs is not supported.

- On tapes written in DataFort format, the encryption switch or blade cannot read and decrypt files with a block size of 1 MB or greater.
- The Top Talker feature is not compatible with redirection zones. The Top Talker feature should not be enabled when an encryption switch or blade is present in the fabric.

Brocade encryption solution overview

The loss of stored private data, trade secrets, intellectual properties, and other sensitive information through theft, or accidental loss of disk or tape media can have widespread negative consequences for governments, businesses, and individuals. This threat is countered by an increasing demand from governments and businesses for solutions that create and enforce policies and procedures that protect stored data. Encryption is a powerful tool for data protection. Brocade provides an encryption solution that resides in a Storage Area Network (SAN) fabric. This location, between computers and storage, is ideal for implementing a solution that works transparently with heterogeneous servers, disk storage subsystems, and tape libraries. Data entering the SAN from a server is encrypted before it is written to storage. When stored data is encrypted, theft or loss of storage media does not pose a security threat.

Figure 2 provides a high-level view of the Brocade encryption solution. Cleartext is sent from the server to the encryption engine, where it is encrypted into ciphertext using one of two encryption algorithms: one for disk storage targets, and one for tape storage targets. The encrypted data cannot be read without first being decrypted. The key management system is required for management of the data encryption keys (DEKs) that are generated by the encryption engine, and used for encrypting and decrypting the data. The key management system is provided by a third-party vendor.

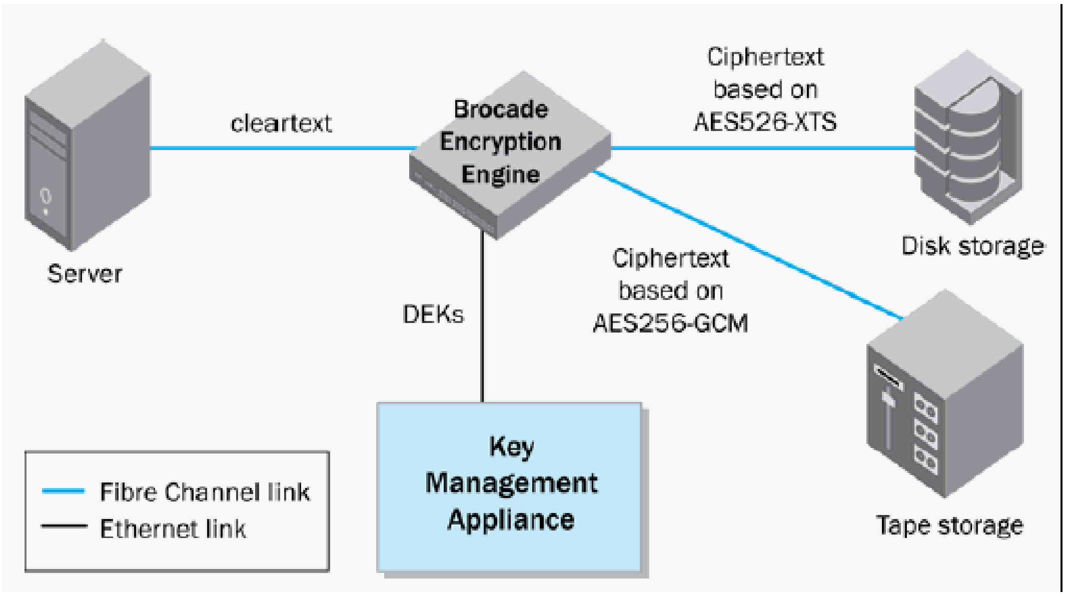


FIGURE 2 **Encryption overview**

Data flow from server to storage

The Brocade Encryption Switch can be introduced into a SAN with minimum disruption, with no need for SAN reconfiguration, and with no need to reconfigure host applications. Frames sent from a host and a target LUN are redirected to a virtual target associated with the encryption switch. The encryption switch then acts as a virtual initiator to forward the frames to the target LUN.

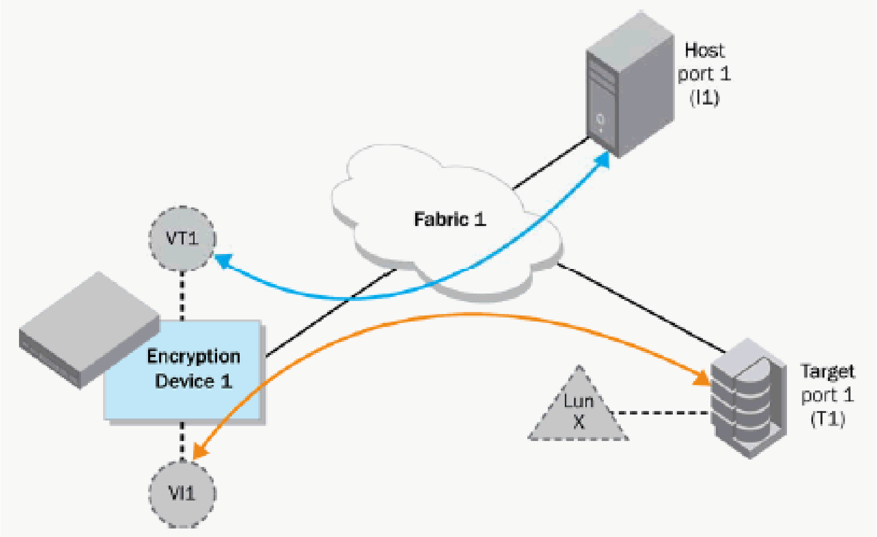


FIGURE 3 Frame redirection

Data encryption key life cycle management

Data encryption keys (DEKs) are generated by the encryption engine. Data is encrypted and decrypted using the same DEK, so a DEK must be preserved at least long enough to decrypt the ciphertext that it created. The length of time data is stored before it is retrieved can vary greatly, and some data may be stored for years or decades before it is accessed. To be sure the data remains accessible, DEKs may also need to be stored for years or decades. Key management systems provide life-cycle management for all DEKs created by the encryption engine. Key management systems are provided by third-party vendors.

Figure 4 shows the relationship of the LAN connections to the key vault and between encryption nodes.

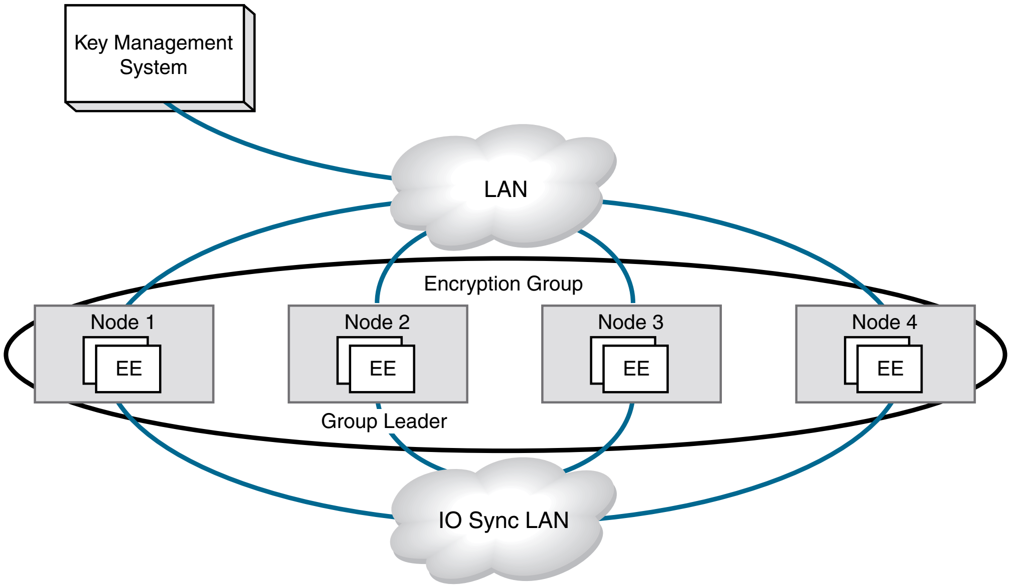


FIGURE 4 **LAN connections to the key vault, and between encryption nodes**

Regardless of the length of the life cycle, there are four stages in the life of a DEK, as shown in Figure 5. A DEK is created by an encryption engine, distributed, then stored in a key vault. The key is used to encrypt and decrypt data at least once, and possibly many times. A DEK may be configured to expire in a certain time frame to avoid becoming compromised. Under those conditions, it must be used one more time to decrypt the data, and the resulting cleartext is encrypted with a new key (rekeyed).

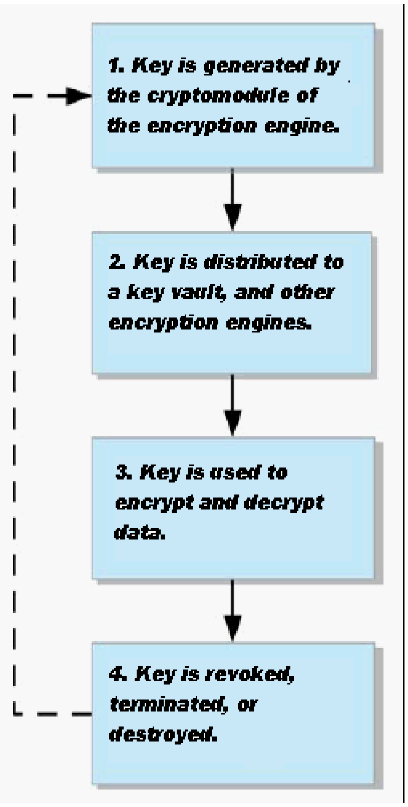

1. Key is generated by the cryptomodule of the encryption engine.
2. Key is distributed to a key vault, and other encryption engines.
3. Key is used to encrypt and decrypt data.
4. Key is revoked, terminated, or destroyed.

FIGURE 5 DEK life cycle

Master key management

Communications with opaque key vaults are encrypted using a master key that is created by the encryption engine on the encryption switch. Currently, this includes the key vaults of all supported key management systems except NetApp LKM.

Master key generation

A master key must be generated by the group leader encryption engine. The master key can be generated once by the group leader, then propagated to the other members of an encryption group.

Master key backup

It is essential to back up the master key immediately after it is generated. The master key may be backed up to any of the following:

- A file as an encrypted key.
- The key management system as an encrypted key record.
- A set of recovery smart cards. This option is available only if the switch is managed by the Brocade Network Advisor (BNA) application (also referred to as the Management application), and if a card reader is available for attachment to the BNA workstation.

  The use of smart cards provides the highest level of security. When smart cards are used, the key is split and written on up to 10 cards. Each card may be kept and stored by a different individual. A quorum of key holders is needed to restore the key. If five key holders exist and the quorum is set to three, then any three of the five key holders is needed to restore the key.

Support for virtual fabrics

The Brocade Encryption Switch does not support the logical switch partitioning capability and, thus, cannot be partitioned, but the switch can be connected to any Logical Switch partition or Logical Fabric using an E_Port.

The FS8-18 Encryption Blades are supported only in a default switch partition. All FS8-18 blades must be placed in a default switch partition in a DCX Backbone chassis. The encryption resource from the default switch partition/fabric can be shared with other logical switch partitions/fabrics or other fabrics only through external device sharing using FCR or EX_Ports through a base switch/fabric. A separate port blade must be used in the base switch/fabric for EX_Port connectivity from the logical switch partition (default switch partition) of FS8-18 blades and host/target fabrics. The EX_Port can be on any external FCR switch.

NOTE
Refer to the *Fabric OS Administrator's Guide* for details on how to configure the Brocade DCX Backbones in virtual fabrics environments, including configuration of default switch partition and any other logical switch partitions.

Cisco Fabric Connectivity support

The Brocade Encryption Switch provides NPIV mode connectivity to Cisco fabrics. Connectivity is supported for Cisco SAN OS 3.3 and later versions.

Cisco fabric connectivity is provided only on the Brocade Encryption Switch. The FS8-18 blade for the Brocade DCX Backbone chassis does not support this feature.

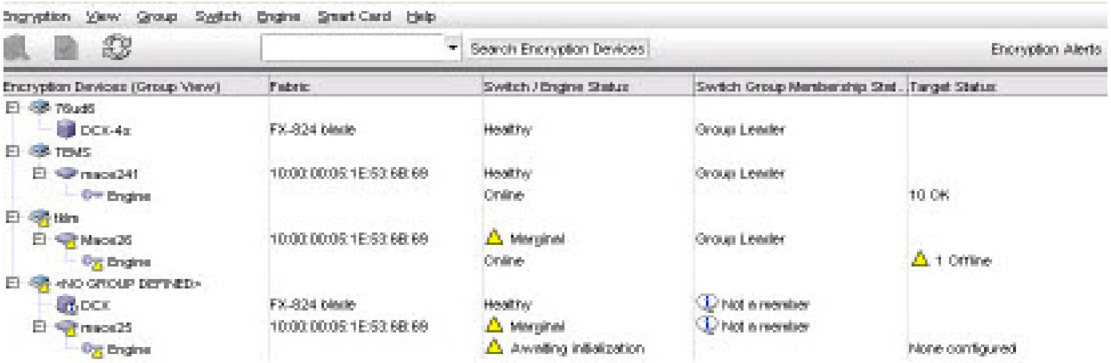
Chapter

Configuring Encryption Using the Management Application 2

In this chapter

Encryption Center features

The **Encryption Center** dialog box is the single launching point for all encryption-related configuration in the Brocade Network Advisor (BNA) Management application (Figure 6). It also provides a table that shows the general status of all encryption-related hardware and functions at a glance. To open the dialog box, select **Configure** > **Encryption**.



Encryption Devices (Group View)	Fabric	Switch / Engine Status	Switch Group Membership Stat...	Target Status
76ud6				
DCX-4s	FX-824 blade	Healthy	Group Leader	
TEMS				
macs241	10:00:00:05:1E:53:6B:69	Healthy	Group Leader	
Engine		Online		10 OK
tiirs				
Macs26	10:00:00:05:1E:53:6B:69	⚠ Marginal	Group Leader	
Engine		Online		⚠ 1 Offline
<NO GROUP DEFINED>				
DCX	FX-824 blade	Healthy	ⓘ Not a member	
macs25	10:00:00:05:1E:53:6B:69	⚠ Marginal	ⓘ Not a member	
Engine		⚠ Awaiting initialization		None configured

FIGURE 6 Encryption Center dialog box

Beginning with Fabric OS 6.4, the Encryption Center is dynamically updated to reflect the latest changes based on any of the following events:

- Encryption group creation or deletion.
- A change in encryption group status or encryption engine status
- Addition or removal of an encryption group member or encryption engine

If you are using the Encryption Center for the first time, please read the following topics before you begin to perform encryption operations:

- "Encryption user privileges" on page 15 describes the Role-based Access Control privileges that are specific to encryption.
- "Smart card usage" on page 16 and the topics that follow describe the options available for the use of Smart Cards for user authentication, system access control, and storing backup copies of data encryption master keys.
- "Network connections" on page 27 describes the network connections that must be in place to enable encryption.
- "Blade processor links" on page 27 describes the steps for interconnecting encryption switches or blades in an encryption group through a dedicated LAN. This must be done before the encryption engines are enabled. Security parameters and certificates cannot be exchanged if these links are not configured and active.
- "Encryption node initialization and certificate generation" on page 28 lists the security parameters and certificates that are generated when an encryption node is initialized.
- "Steps for connecting to a DPM appliance" on page 29 lists the supported key manager appliances, and lists topics that provide additional detail.

Encryption user privileges

In BNA, resource groups are assigned privileges, roles, and fabrics. Privileges are not directly assigned to users; users get privileges because they belong to a role in a resource group. A user can only belong to one resource group at a time.

BNA provides three pre-configured roles:

- Storage encryption configuration
- Storage encryption key operations
- Storage encryption security

Table 1 lists the associated roles and their read/write access to specific operations. The functions are enabled from the **Encryption Center** dialog box:

TABLE 1 Encryption privileges

Privilege	Read/Write
Storage Encryption Configuration	• Launch the Encryption center dialog box. • View switch, group, or engine properties. • View the Encryption Group Properties Security tab. • View encryption targets, hosts, and LUNs. • View LUN centric view • View all rekey sessions • Add/remove paths and edit LUN configuration on LUN centric view • Rebalance encryption engines. • Clear tape LUN statistics • Create a new encryption group or add a switch to an existing encryption group. • Edit group engine properties (except for the Security tab) • Add targets. • Select encryption targets and LUNs to be encrypted or edit LUN encryption settings. • Edit encryption target hosts configuration. • Show tape LUN statistics.
Storage Encryption Key Operations	• Launch the Encryption center dialog box. • View switch, group, or engine properties, • View the Encryption Group Properties Security tab. • View encryption targets, hosts, and LUNs. • View LUN centric view. • View all rekey sessions. • Initiate manual rekeying of all disk LUNs. • Initiate refresh DEK. • Enable and disable an encryption engine. • Decommission LUNs. • Zeroize an encryption engine. • Restore a master key. • Edit key vault credentials. • Show tape LUN statistics.

TABLE 1 Encryption privileges (Continued)

Privilege	Read/Write
Storage Encryption Security	• Launch the Encryption center dialog box. • View switch, group, or engine properties. • View Encryption Group Properties Security tab. • View LUN centric view. • View all rekey sessions. • View encryption targets, hosts, and LUNs. • Create a master key. • Backup a master key. • Edit smart card. • View and modify settings on the Encryption Group Properties Security tab (quorum size, authentication cards list and system card requirement). • Establish link keys for LKM key managers. • Show tape LUN statistics.

Smart card usage

Smart Cards are credit card-sized cards that contain a CPU and persistent memory. Smart cards can be used as security devices. You must have *Storage Encryption Security* user privileges to activate, register, and configure smart cards.

Smart cards can be used to do the following:

- Control user access to BNA security administrator roles
- Control activation of encryption engines
- Securely store backup copies of master keys

Smart card readers provide a plug-and-play interface that allows you to read and write to a smart card. The following smart card readers are supported:

- GemPlus GemPC USB

  http://www.gemalto.com/readers/index.html

- SCM MicrosystemsSCR331

  http://www.scmmicro.com/security/view_product_en.php?PID=2

NOTE
Only the Brocade smart cards that are included with the encryption switches are supported.

Using authentication cards with a card reader

When authentication cards are used, one or more authentication cards must be read by a card reader attached to a Management application workstation to enable certain security-sensitive operations. These include the following:

- Performing master key generation, backup, and restore operations.
- Registering or deregistering and replacement of authentication cards.
- Enabling and disabling the use of system cards.
- Changing the quorum size for authentication cards.

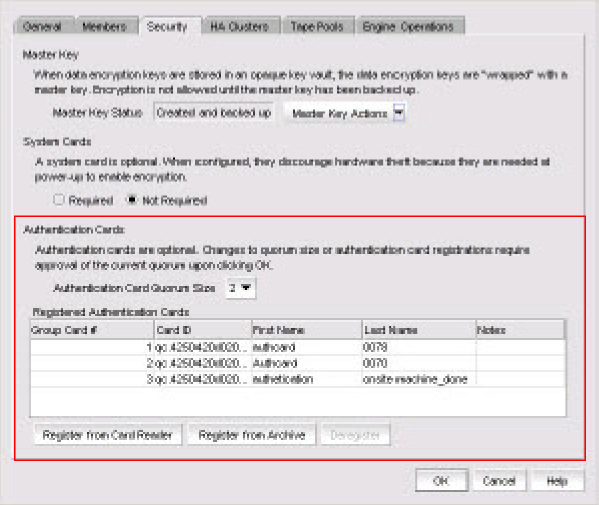
- Establishing a trusted link with the NetApp LKM key vault.
- Decommissioning a LUN.

When a quorum of authentication cards is registered for use, authentication must be provided before you are granted access.

Registering authentication cards from a card reader

To register an authentication card or a set of authentication cards from a card reader, have the cards physically available. Authentication cards can be registered during encryption group or member configuration when running the configuration wizard, or they can be registered using the following procedure.

1. Select **Configure > Encryption** from the menu task bar to display the **Encryption Center** dialog box (Refer to Figure 6 on page 14).
2. Select an encryption group from the **Encryption Center Devices** table, then select **Group > Security** from the menu task bar to display the **Encryption Group Properties** dialog box. The **Security** tab is selected (Figure 7).



Master Key

When data encryption keys are stored in an opaque key vault, the data encryption keys are "wrapped" with a master key. Encryption is not allowed until the master key has been backed up.

Master Key Status: Created and backed up

System Cards

A system card is optional. When configured, they discourage hardware theft because they are needed at power-up to enable encryption.

○ Required ● Not Required

Authentication Cards

Authentication cards are optional. Changes to quorum size or authentication card registrations require approval of the current quorum upon clicking OK.

Authentication Card Quorum Size: 2

Registered Authentication Cards

Group Card #	Card ID	First Name	Last Name	Notes
1	gc.4250420d020...	authcard	0078	
2	gc.4250420d020...	Authcard	0070	
3	gc.4250420d020...	authetication	onsite machine_done	

FIGURE 7 Encryption Group Properties dialog box - registering authentication cards

The dialog box contains the following information:

- **Group Card#**: A number assigned to the card as it is registered.
- **Card ID**: The serial number read from the smart card.
- **First Name**: The first name of the person assigned to the card.
- **Last Name**: The last name of the person assigned to the card.
- **Notes**: An optional entry of information.
- **Register from Card Reader** button: Launches the **Add Authentication Card** dialog box.
- **Register from Archive** button: Launches the **Add Authentication Card** dialog box.
- **Deregister** button: Deregisters a card selected from the **Registered Authentication Cards** table, which enables the cards to be removed from the switch and the database.

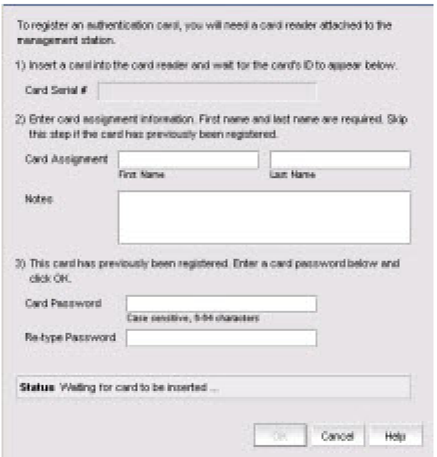
3. Locate the **Authentication Card Quorum Size** and select the quorum size from the list.

   The quorum size is the minimum number of cards necessary to enable the card holders to perform the security sensitive operations listed above. The maximum quorum size is five cards. The actual number of authentication cards registered is always more than the quorum size, so if you set the quorum size to five, for example, you will need to register at least six cards in the subsequent steps.

   NOTE
   Ignore the **System Cards** setting for now.

4. Click **Register from Card Reader** to register a new card.

   The **Add Authentication Card** dialog box displays (Figure 8).



To register an authentication card, you will need a card reader attached to the management station.

1) Insert a card into the card reader and wait for the card's ID to appear below.

Card Serial #

2) Enter card assignment information. First name and last name are required. Skip this step if the card has previously been registered.

Card Assignment

First Name

Last Name

Notes

3) This card has previously been registered. Enter a card password below and click OK.

Card Password

Case sensitive, 5-64 characters

Re-type Password

Status Waiting for card to be inserted ...

FIGURE 8 Add Authentication Card dialog box

The dialog box contains the following information:

- **Card Serial#:** A serial number read from the smart card.
- **Card Assignment:** The first and last name of the person assigned to the card.
- **Notes:** An optional entry of information.
- **Card Password:** Create a password for the card holder to enter for user verification.
- **Re-type Password:** Re-enter the password in this field.
- **Status:** Indicates the status when a card is being registered.

5. Insert a smart card into the card reader. Wait for the card serial number to appear, enter card assignment information as directed, then click **OK**.

6. Wait for the confirmation dialog box indicating initialization is done, then click **OK**.

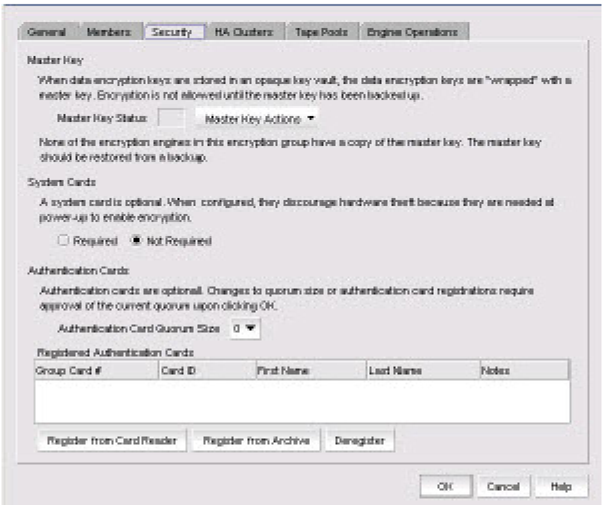
   The card is added to the **Registered Authentication Cards** table.

7. Repeat step 5 through step 6 until you have successfully registered all cards. Ensure that the number of cards registered equals at least the quorum size plus one.

Registering authentication cards from the database

Smart cards that are already in the Management program's database can be registered as authentication cards.

1. Select **Configure** > **Encryption** from the menu task bar to display the **Encryption Center** dialog box (Refer to Figure 6 on page 14).
2. Select an encryption group from the **Encryption Center Devices** table, then select **Group** > **Security** from the menu task bar to display the **Encryption Group Properties** dialog box. The **Security** tab is selected (Figure 9).



When data encryption keys are stored in an opaque key vault, the data encryption keys are "wrapped" with a master key. Encryption is not allowed until the master key has been backed up.

None of the encryption engines in this encryption group have a copy of the master key. The master key should be restored from a backup.

A system card is optional. When configured, they discourage hardware theft because they are needed at power-up to enable encryption.

Authentication cards are optional. Changes to quorum size or authentication card registrations require approval of the current quorum upon clicking OK.

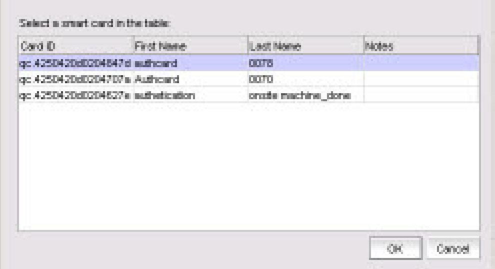
Authentication Card Quorum Size: 0

Group Card #	Card ID	First Name	Last Name	Notes

FIGURE 9 **Encryption Group Properties dialog box - Security tab**

3. Click **Register from Archive.**

   The **Authentication Cards** dialog box displays (Figure 10). The table lists the smart cards that are in the database.



Select a smart card in the table:

Card ID	First Name	Last Name	Notes
qc 4250420d0204847d	authcard	0078	
qc 4250420d0204707a	Authcard	0070	
qc 4250420d0204627a	authetication	oracle machine_done	

FIGURE 10 **Authentication Cards dialog box - registering smart cards from archive**

4. Select a card from the table, then click **OK.**
5. Wait for the confirmation dialog box indicating initialization is done, then click **OK.**

   The card is added to the **Registered Authentication Cards** table.

Deregistering an authentication card

Authentication cards can be removed from the database and the switch by deregistering them. Complete the following procedure to deregister an authentication card.

1. Select **Configure > Encryption** from the menu task bar to display the **Encryption Center** dialog box (Refer to Figure 6 on page 14).
2. Select an encryption group from the **Encryption Center Devices** table, then select **Group > Security** from the menu task bar to display the **Encryption Group Properties** dialog box. The **Security** tab is selected.
3. Select the desired authentication card in the **Registered Authentication Cards** table, then click **Deregister**.
4. Click **Yes** to confirm deregistration.

   The registered authentication card is removed from the table.
5. Click **OK**.

   The card is deregistered from the group.

Setting a quorum for authentication cards

To authenticate using a quorum of authentication cards, complete the following steps:

1. When using the **Authenticate** dialog box, gather the number of cards needed according to the instructions in the dialog box. The registered cards and the assigned owners are listed in the table near the bottom of the dialog box.

   The dialog box contains the following information:

   - **Card ID**: Insert a smart card into an attached card reader, and wait for the card ID to appear in this field.
   - **Password**: The card holder must enter a password for the card.
   - **Authenticate** button: Authenticates the card after entering the password.
   - **Currently registered authentication cards** table: Lists the currently registered cards, showing the card ID and the name of the person assigned to the card.
   - **Status**: Displays the status of the card authentication operation.
2. Insert a card, then wait for the ID to appear in the **Card ID** field.
3. Enter the assigned password, then click **Authenticate**.
4. Wait for the confirmation dialog box, then click **OK**.
5. Repeat step 2 through step 4 for each card until at least the quorum plus one is reached, then click **OK**.

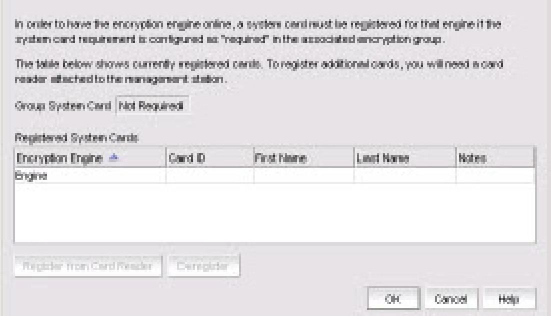
Using system cards

System cards are smart cards that can be used to control activation of encryption engines. You can choose whether the use of a system card is required or not. Encryption switches and blades have a card reader that enables the use of a system card. System cards discourage theft of encryption switches or blades by requiring the use of a system card at the switch or blade to enable the encryption engine after a power off.

When the switch or blade is powered off, the encryption engine will not work without first inserting a system card into its card reader. If someone removes a switch or blade with the intent of accessing the encryption engine, it will function as an ordinary FC switch or blade when it is powered up, but use of the encryption engine is denied.

To register a system card from a card reader, the smart card must be physically available.

The **System Cards** dialog box can be accessed by selecting a switch from the **Encryption Center Devices** table, then selecting **Switch > System Cards** from the menu task bar. The **Register System Card** dialog box displays (Figure 11).



In order to have the encryption engine online, a system card must be registered for that engine if the system card requirement is configured as "required" in the associated encryption group.

The table below shows currently registered cards. To register additional cards, you will need a card reader attached to the management station.

Group System Card: Not Required

Registered System Cards

Encryption Engine	Card ID	First Name	Last Name	Notes
Engine				

FIGURE 11 System Cards dialog box

The dialog box contains the following information:

- **Group System Card**: Identifies if smart cards are used to control activation of encryption engines.
- **Registered System Cards** table: Lists all currently registered system card serial numbers and to whom the cards are assigned by first and last name. Also included are any free-form notes related to the cards.
- **Register from Card Reader** button: Launches the **Register from Card Reader** dialog box.
- **Deregister** button: Launches the **Deregister** dialog box.

Enabling or disabling the system card requirement

To use a system card to control activation of an encryption engine on a switch, you must enable the system card requirement. If a system card is required, it must be read by the card reader on the switch. You access the system card GUI from the Security tab.

Complete the following procedure to enable or disable the system card requirement.

1. Select **Configure** > **Encryption** from the menu task bar to display the **Encryption Center** dialog box (Refer to Figure 6 on page 14).
2. Select a switch from the **Encryption Center Devices** table, then select **Switch** > **System Cards** from the menu task bar.

   The **System Cards** dialog box displays (Refer to Figure 11 on page 21).
3. Do one of the following:
   - Set **System Cards** to **Required** to require the use of a system card for controlling activation of the encryption engine.
   - Set **System Cards** to **Not Required** to permit activation of the encryption engine without the need to read a system card first.
4. Click **OK**.

Registering systems card from a card reader

To register a system card from a card reader, a smart card must be physically available. System cards can be registered during encryption group creation or member configuration when running the configuration wizard, or they can be registered using the following procedure:

1. Select **Configure** > **Encryption** from the menu task bar to display the **Encryption Center** dialog box (Refer to Figure 6 on page 14).
2. Select a switch from the **Encryption Center Devices** table that is not already in an encryption group, then select **Switch** > **System Cards** from the menu task bar.

   The **System Cards** dialog box displays (Refer to Figure 11 on page 21). The **Registered System Cards** table lists all currently registered system card serial numbers and to whom they are assigned. Also included are any notes related to the cards.
3. Click **Register from Card Reader.**
4. Insert a smart card into the card reader.
5. Wait for the card serial number to appear, then enter card assignment information as directed and click **OK**.
6. Wait for the confirmation dialog box indicating initialization is done, then click **OK**.

   The card is added to the **Registered System Cards** table.

NOTE
Store the card in a secure location, not in proximity to the switch or blade.

Deregistering system cards

System cards can be removed from the database by deregistering them. Use the following procedure to deregister a system card:

1. Select **Configure** > **Encryption** from the menu task bar to display the **Encryption Center** dialog box. (Refer to Figure 6 on page 14.)
2. Select the switch from the **Encryption Center Devices** table, then select **Switch** > **System Cards** from the menu task bar.

   The **System Cards** dialog box displays. (Refer to Figure 11 on page 21.)
3. Select the system card to deregister, then click **Deregister**.
4. A confirmation dialog box displays. Click **OK** to confirm deregistration.

   The card is removed from the **Registered System Cards** table.

Using smart cards

Smart cards can be used for user authentication, master key storage and backup, and as a system card for authorizing use of encryption operations. Card types identify if the smart card is a system card, authentication card, or recovery set.

The Smart Card Asset Tracking dialog box displays two tables: **Smart Cards** table and **Card Details** table.

- Selecting an authentication in the **Smart Cards** table, displays all group names for which the card is registered in the **Card Details** table.
- Selecting a system cards in the **Smart Cards** table displays all encryption engines for which the card is registered by switch name and, for encryption blades, slot number in the **Card Details** table.
- Selecting a recovery card in the **Smart Cards** table displays, the group name, the card creation date, and the position of the card in the set (for example, Card 1 of 3) in the **Card Details** table.

Tracking smart cards

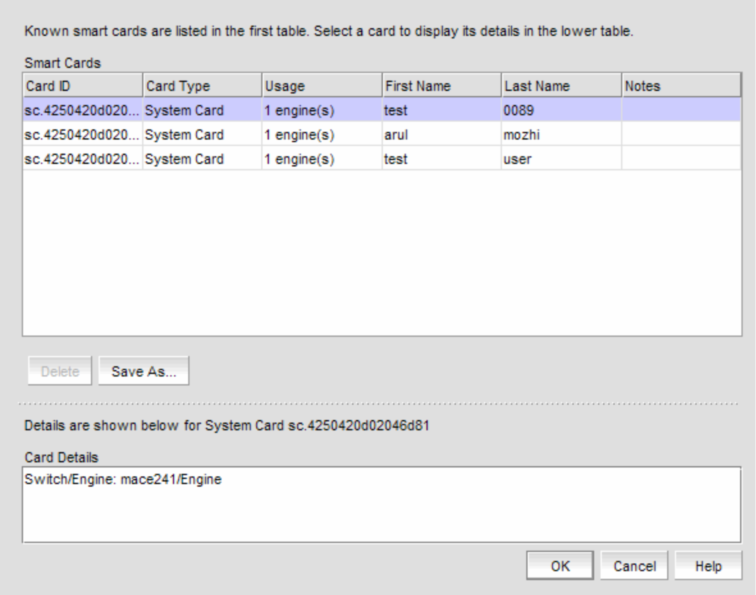
1. Select **Configure** > **Encryption** from the menu task bar to display the **Encryption Center** dialog box. (Refer to Figure 6 on page 14.)
2. Select **Smart Card** > **Smart Card Tracking** from the menu task bar to display the **Smart Card Asset Tracking** dialog box (Figure 12).

   The **Smart Cards** table lists the known smart cards and the details for the smart cards. These details include the following:

   - **Card ID**: Lists the smart card ID, prefixed with an ID that identifies how the card id used. For example, rc.123566b700017818, where rc stands for recovery card.
   - **Card Type**: Options are: System card, Authentication card, and Recovery set.
   - **Usage**: Usage content varies based on the card type.
     - For Authentication cards, the **Usage** column shows the number of groups for which the card is registered.
     - For System cards, the **Usage** column shows the number of encryption engines for which the card is registered.
     - For Recovery cards, the **Usage** column shows the group name and the creation date.
   - **First Name**: The first name of the person (up to 64 characters) to whom the smart card is assigned. All characters are valid in the editable columns, including spaces. Editing these values in the BNA application does not modify the information that is stored on the card.
   - **Last Name**: The last name of the person (up to 64 characters) to whom the smart card is assigned. All characters are valid in the editable columns, including spaces. Editing these values in the BNA application does not modify the information that is stored on the card.
   - **Notes**: Miscellaneous notes (up to 256 characters) related to the smart card. Editing these values in the BNA does not modify the information that is stored on the card. Notes are optional.
   - **Delete** button: Deletes a selected smart card from the BNA database.

   NOTE
   You can remove smart cards from the table to keep the Smart Cards table at a manageable size, but removing the card from the table does not invalidate it; the smart card can still be used.

   - **Save As** button: Saves the entire list of smart cards to a file. The available formats are comma-separated values (.csv) and HTML (.html).
   - **Card Details** table: Card details vary based on the card type.
     - For Authentication cards, the **Card Details** table shows all group names for which the card is registered.
     - For System cards, the **Card Details** table shows all encryption engines for which the card is registered by switch name and, for encryption blades, slot number.
     - For Recovery cards, the **Card Details** table shows the group name, the card creation date, and the position of the card in the set (for example, Card 1 of 3).



Known smart cards are listed in the first table. Select a card to display its details in the lower table.

Smart Cards

Card ID	Card Type	Usage	First Name	Last Name	Notes
sc.4250420d020...	System Card	1 engine(s)	test	0089	
sc.4250420d020...	System Card	1 engine(s)	arul	mozhi	
sc.4250420d020...	System Card	1 engine(s)	test	user	

Details are shown below for System Card sc.4250420d02046d81

Card Details

Switch/Engine: mace241/Engine

FIGURE 12 Smart Card asset tracking dialog box

3. Select a smart card from the table, then do one of the following:

   - Click **Delete** to remove the smart card from the BNA database. Deleting smart cards from the BNA database keeps the **Smart Cards** table at a manageable size, but does not invalidate the smart card. The smart card can still be used. You must deregister a smart card to invalidate its use.

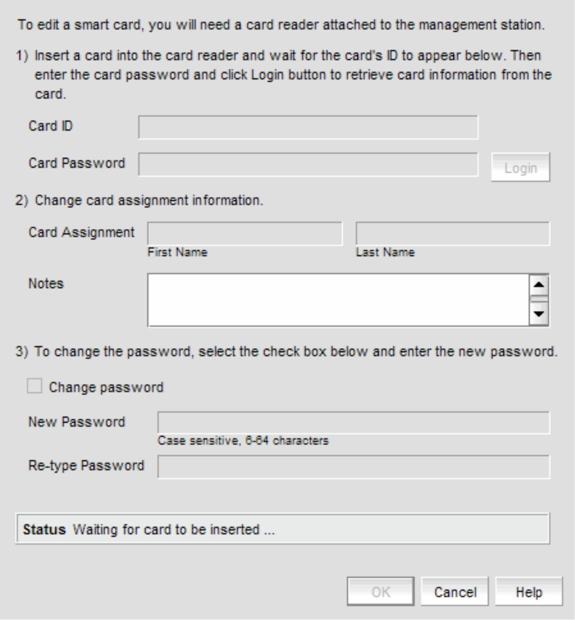
     NOTE
     The Delete operation applies only to recovery cards.

   - Click **Save As** to save the entire list of smart cards to a file. The available formats are comma-separated values (.csv) and HTML files (.html).

Editing smart cards

Smart cards can be used for user authentication, master key storage and backup, and as a system card for authorizing use of encryption operations.

1. From the **Encryption Center** dialog box, select **Smart Card** > **Edit Smart Card** from the menu task bar to display the **Edit Smart Card** dialog box (Figure 13).



To edit a smart card, you will need a card reader attached to the management station.

1) Insert a card into the card reader and wait for the card's ID to appear below. Then enter the card password and click Login button to retrieve card information from the card.

2) Change card assignment information.

3) To change the password, select the check box below and enter the new password.

□ Change password

Case sensitive, 6-64 characters

Status Waiting for card to be inserted ...

FIGURE 13 Edit Smart Card dialog box

2. Insert the smart card into the card reader.
3. After the card's ID is displayed by the card reader in the **Card ID** field, enter the security administrator password used to allow editing of the smart card, then click **Login**.

   NOTE
   The **Card Password** field is activated after the card ID is read, and the **Login** button is activated after the password is entered in the **Card Password** field.

4. Edit the card as needed. Note the following:
   - **Card Assignment**: A maximum of 64 characters is permitted for the user first and last name to whom the card is assigned. All characters are valid in the editable columns, including spaces.
   - **Notes**: A maximum of 256 characters is permitted for any miscellaneous notes. Editing these values in the BNA application does not modify the information that is stored on the card. Notes are optional.
   - The **Change Password** check box must be selected before you can enter the new password information. You must re-enter the new password for verification.
5. Click **OK**.

NOTE
You can view the status indicator at the bottom of the dialog box to determine card reader status.

Network connections

Before you use the encryption setup wizard for the first time, you must have the following required network connections:

- The management ports on all encryption switches and 8-slot Backbone Chassis CPs that have encryption blades installed must have a LAN connection to the SAN management program, and must be available for discovery.
- A supported key management appliance must be connected on the same LAN as the management port of the encryption switches, 8-slot Backbone Chassis CPs, and the SAN Management program.
- In some cases, you might want to have an external host available on the LAN to facilitate certificate exchange between encryption nodes and the key management appliance. You may use the SAN management program host computer rather than an external host.
- All switches in the planned encryption group must be interconnected on a private LAN. This LAN is used to exchange security parameters and certificates, and to synchronize encryption engine operations.

Blade processor links

Each encryption switch or blade has two GbE ports labeled Ge0 and Ge1. The Ge0 and Ge1 ports are Ethernet ports that connect encryption switches and blades to other encryption switches and blades. Both ports of each encryption switch or blade must be connected to the same IP network and the same subnet. Static IP addresses should be assigned. Neither VLANs nor DHCP should be used. These two ports are bonded together as a single virtual network interface to provide link layer redundancy.

All encryption switches and blades in an encryption group must be interconnected by these links through a dedicated LAN before their encryption engines are enabled. Both ports of each encryption switch or blade must be connected to the same IP network and the same subnet. Static IP addresses should be assigned. VLANs should not be used, and DHCP should not be used. Security parameters and certificates cannot be exchanged if these links are not configured and active.

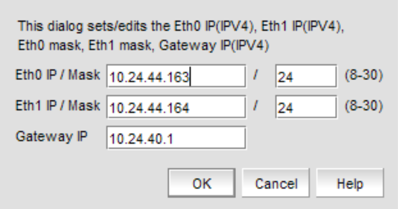
The **Blade Processor Link** dialog box can be launched from the following locations:

- Select an encryption group from the **Encryption Center Devices** table, then select **Group > HA Clusters** from the menu task bar. The **Properties** dialog box displays with the **HA Clusters** tab selected. Select a device from the **Non-HA Encryption Engines** table, then click **Configure Blade Processor Link.**
- Select a group, switch, or engine from the **Encryption Center Devices** table, then select **Group/Switch/Engine > Targets** from the menu task bar. Select a container from the **Encryption Targets** table, click **LUNs**, then click **Configure Blade Processor Link.**
- Select an engine from the **Encryption Center Devices** table, then select **Engine > Blade Processor Link.**

Configuring blade processor links

To configure blade processor links, complete the following steps:

1. Select **Configure > Encryption** from the menu task bar to display the **Encryption Center** dialog box. (Refer to Figure 6 on page 14.)
2. Select the encryption engine from the **Encryption Center Devices** table, then select **Engine > Blade Processor Link** from the menu task bar to display the **Blade Processor Link** dialog box (Figure 14).



This dialog sets/edits the Eth0 IP(IPV4), Eth1 IP(IPV4), Eth0 mask, Eth1 mask, Gateway IP(IPV4)

Eth0 IP / Mask 10.24.44.163 / 24 (8-30)

Eth1 IP / Mask 10.24.44.164 / 24 (8-30)

Gateway IP 10.24.40.1

FIGURE 14 Blade Processor Link dialog box

3. Enter the link IP address and mask, and the gateway IP address.
   - **Eth0 IP /Mask** identifies the Ge0 interface IP address and mask.
   - **Eth1 IP /Mask** identifies the Ge1 interface IP address and mask.
   - The **Gateway IP** address is optional.
4. Click **OK.**

Encryption node initialization and certificate generation

When an encryption node is initialized, the following security parameters and certificates are generated:

- FIPS crypto officer
- FIPS user
- Node CP certificate
- A signed Key Authentication Center (KAC) certificate
- A KAC Certificate Signing Request (CSR)

From the standpoint of external SAN management application operations, the FIPS crypto officer, FIPS user, and node CP certificates are transparent to users. The KAC certificates are required for operations with key managers. In most cases, KAC certificate signing requests must be sent to a Certificate Authority (CA) for signing to provide authentication before the certificate can be used. In all cases, signed KACs must be present on each switch.

Setting encryption node initialization

Encryption nodes are initialized by the **Configure Switch Encryption** wizard when you confirm a configuration. Encryption nodes may also be initialized from the **Encryption Center** dialog box.

1. Select a switch from the **Encryption Center Devices** table, then select **Switch** > **Init Node** from the menu task bar.
2. Select **Yes** after reading the warning message to initialize the node.

Steps for connecting to a DPM appliance

All switches that you plan to include in an encryption group must have a secure connection to the RSA Data Protection Manager (DPM). The following is a suggested order of steps needed to create a secure connection to the DPM.

NOTE
The Brocade Encryption Switch uses the manual enrollment of identities with client registration to connect with DPM 3.x servers. Client registration is done automatically when you upgrade to Fabric 7.1.0 from an earlier Fabric OS version; no user interaction is required.

Once completed, client registration occurs after key vault registration, when the Brocade Encryption Switch attempts to connect to the DPM server for the first time.

1. Export the KAC CSR to a location accessible to a CA for signing. Refer to “Exporting the KAC certificate signing request (CSR)” on page 30.
2. Submit the KAC CSR for signing by a CA. Refer to “Submitting the CSR to a certificate authority” on page 30.
3. Set the KAC certificate registration expiry. Refer to “KAC certificate registration expiry” on page 31.
4. Import the signed certificate into the Fabric OS encryption node. Refer to “Importing the signed KAC certificate” on page 31.
5. Upload the signed KAC and CA certificates onto the DPM appliance and select the appropriate key classes. Refer to the following:
   - “Uploading the CA certificate onto the DPM appliance (and first-time configurations)” on page 32
   - “Uploading the KAC certificate onto the DPM appliance (manual identity enrollment)” on page 33
6. If dual DPM appliances are used for high availability, the DPM appliances must be clustered, and must operate in maximum availability mode, as described in the DPM appliance user documentation. Refer to “DPM key vault high availability deployment” on page 33.

Exporting the KAC certificate signing request (CSR)

1. Export the KAC CSR to a temporary location prior to submitting the KAC CSR to a CA for signing.
2. Synchronize the time on the switch and the key manager appliance. Time settings should be within one minute of each other. Differences in time can invalidate certificates and cause key vault operations to fail.
3. Select a switch from the **Encryption Center Devices** table, then select **Switch** > **Properties** from the menu task bar to display the **Properties** dialog box.

   NOTE
   You can also select a switch from the **Encryption Center Devices** table, then click the **Properties** icon.

4. Do one of the following:
   - If a CSR is present, click **Export**.
   - If a CSR is not present, select a switch from the **Encryption Center Devices** table, then select **Switch** > **Init Node** from the menu task bar. This generates switch security parameters and certificates, including the KAC CSR.
5. Save the file. The default location for the exported file is in the **Documents** folder.

NOTE
The CSR is exported in Privacy Enhanced Mail (.pem) format. This is the format required in exchanges with Certificate Authorities (CAs).

Submitting the CSR to a certificate authority

The CSR must be submitted to a Certificate Authority (CA) to be signed. The CA is a trusted third-party entity that signs the CSR. Several CAs are available and procedures vary, but the general steps are as follows:

1. Open an SSL/TLS connection to an X.509 server.
2. Submit the CSR for signing.
3. Request the signed certificate.

   Generally, a public key, the signed KAC certificate, and a signed CA certificate are returned.

4. Download and store the signed certificates.

The following example submits a CSR to the demoCA from RSA:

```
cd /opt/CA/demoCA
openssl x509 -req -sha1 -CAcreateserial -in certs/<Switch CSR Name> -days 365
-CA cacert.pem -CAkey private/cakey.pem -out newcerts/<Switch Cert Name>
```

NOTE
You can change the number of days that a certificate will expire based on your site's security policies. For more information on changing the certificate expiry date, refer to "KAC certificate registration expiry" on page 31.

KAC certificate registration expiry

It is important to keep track as to when your signed KAC certificates will expire. Failure to work with valid certificates causes certain commands to not work as expected. If you are using the certificate expiry feature and the certificate expires, the key vault server will not respond as expected. For example, the group leader in an encryption group might show that the key vault is connected; however, a member node reports that the key vault is not responding.

To verify the certificate expiration date, use the following command:

```
openssl x509 –in newcerts/<Switch Cert Name> -dates –noout

Output:
            Not Before: Dec  4 18:03:14 2009 GMT
            Not After : Dec  4 18:03:14 2010 GMT
```

In the example above, the certificate validity is active until “Dec 4 18:03:14 2010 GMT.” After the KAC certificate has expired, the registration process must be redone.

NOTE
In the event that the signed KAC certificate must be re-registered, you will need to log in to the key vault web interface and upload the new signed KAC certificate for the corresponding Brocade Encryption Switch identity.

You can change the value of the certificate expiration date using the following command:

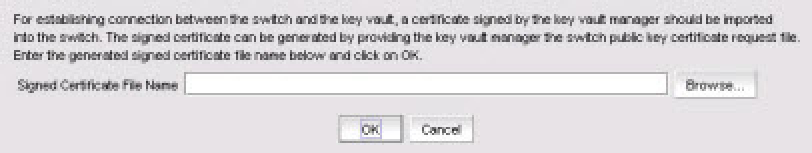
```
openssl x509 -req -sha1 -CAcreateserial -in certs/<Switch CSR Name> -days 365 -CA
cacert.pem -CAkey private/cakey.pem -out newcerts/<Switch Cert Name>
```

In the example above, the certificate is valid for a period of one year (365 days). You can increase or decrease this value according to your own specific needs. The default is 3649 days, or 10 years.

Importing the signed KAC certificate

After a KAC CSR has been submitted and signed by a CA, the signed certificate must be imported into the switch.

1. Select a switch from the **Encryption Center Devices** table, then select **Switch** > **Import Certificate** from the menu task bar to display the **Import Signed Certificate** dialog box (Figure 15).



For establishing connection between the switch and the key vault, a certificate signed by the key vault manager should be imported into the switch. The signed certificate can be generated by providing the key vault manager the switch public key certificate request file. Enter the generated signed certificate file name below and click on OK.

Signed Certificate File Name

FIGURE 15 **Import Signed Certificate dialog box**

2. Browse to the location where the signed certificate is stored, then click **OK**.

   The signed certificate is stored on the switch.

Uploading the CA certificate onto the DPM appliance (and first-time configurations)

After an encryption group is created, you need to install the signing authority certificate (CA certificate) onto the DPM appliance.

1. Open a web browser and connect to the DPM appliance setup page. You will need the URL and have the proper authority level, user name, and password.
2. Select the Operations tab.
3. Select Certificate Upload.
4. In the SSLCAcertificateFile field, enter the full local path of the CA certificate. Do not use the UNC naming convention format.
5. Select Upload, Configure SSL, and Restart Webserver.
6. After the web server restarts, enter the root password.
7. Open another web browser window, and start the RSA management user interface.

   You will need the URL, and have the proper authority level, user name, and password.

   NOTE
   The Identity Group name used in the next step might not exist in a freshly installed DPM. To establish an Identity Group name, click the Identity Group tab, and create a name. The name Hardware Retail Group is used as an example in the following steps.

8. Select the Key Classes tab. The key classes must be created only once, regardless of the number of nodes in your encryption group or the number of encryption groups that will be sharing this DPM.

   kcn.1998-01.com.brocade:DEK_AES_256_XTS

   kcn.1998-01.com.brocade:DEK_AES_256_CCM

   kcn.1998-01.com.brocade:DEK_AES_256_GCM

   kcn.1998-01.com.brocade:DEK_AES_256_ECB

   a. Click Create.
   b. Type the key name string into the Name field.
   c. Select Hardware Retail Group for Identity Group.
   d. Deselect Activated Keys Have Duration.
   e. Select AES for Algorithm.
   f. Select 256 for Key Size.
   g. Select the Mode for the respective key classes as follows:

      XTS for Key Class "kcn.1998-01.com.brocade:DEK_AES_256_XTS"

      CBC for Key Class "kcn.1998-01.com.brocade:DEK_AES_256_CCM"

      CBC for Key Class "kcn.1998-01.com.brocade:DEK_AES_256_GCM"

      ECB for Key Class "kcn.1998-01.com.brocade:DEK_AES_256_ECB"

h. Click **Next**.

i. Repeat step a through step h for each key class.

j. Click **Finish**.

Uploading the KAC certificate onto the DPM appliance (manual identity enrollment)

NOTE
The Brocade Encryption Switch will not use the Identity Auto Enrollment feature supported with DPM 3.x servers. You must complete the identity enrollment manually to configure the DPM 3.x server with the Brocade Encryption Switch as described in this section.

You need to install the switch public key certificate (KAC certificate). For each encryption node, manually create an identity as follows:

1. Select the **Identities** tab.
2. Click **Create**.
3. Enter a label for the node in the **Name** field. This is a user-defined identifier.
4. Select the **Hardware Retail Group** in the **Identity Groups** field.
5. Select the **Operational User** role in the **Authorization** field.
6. Click **Browse** and select the imported certificate as the **Identity certificate**.
7. Click **Save**.

The CA certificate file referenced in the **SSLCAcertificateFile** field (see step 4) must be imported and registered on the switch designated as an encryption group leader. You might want to note this location before proceeding to "Loading the CA certificate onto the encryption group leader" on page 34.

DPM key vault high availability deployment

When dual DPM appliances are used for high availability, the DPM appliances must be clustered and must operate in maximum availability mode, as described in the DPM appliance user documentation.

When dual DPM appliances are clustered, they are accessed using an IP load balancer. For a complete high availability deployment, the multiple IP load balancers are clustered, and the IP load balancer cluster exposes a virtual IP address called a floating IP address. The floating IP address must be registered on the encryption group leader.

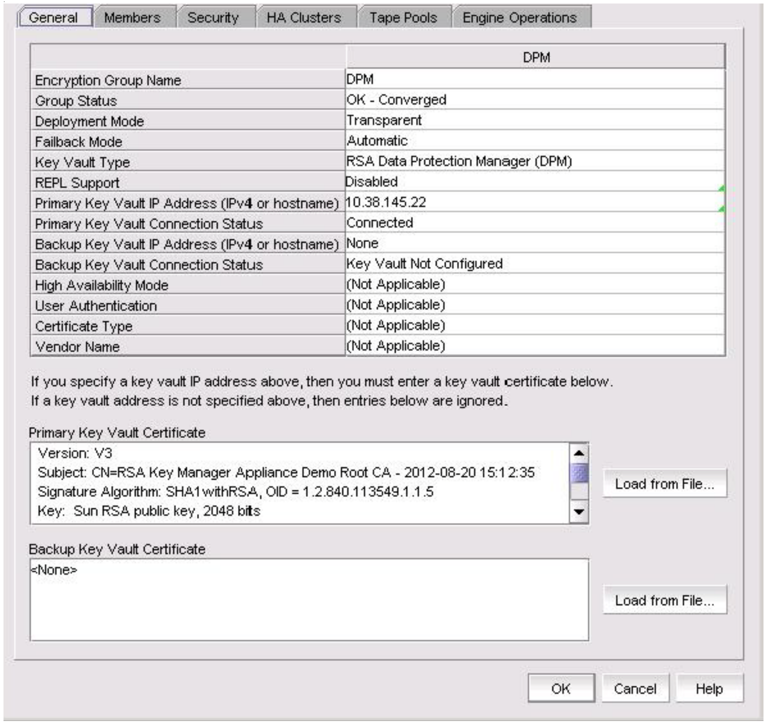
Neither the secondary DPM appliance nor individual DPM appliance IP addresses should be registered.

Loading the CA certificate onto the encryption group leader

The certificate for the CA that signed the switch KAC CSRs must be loaded onto the encryption group leader. The group leader can then distribute the CA certificate to the encryption group members.

1. From the **Encryption Center**, select a group from the **Encryption Center Devices** table, then select **Group** > **Properties** from the menu task bar to display the **Encryption Group Properties** dialog box. The **General** tab is selected (Figure 16).

   If groups are not visible in the **Encryption Devices** table, select **View** > **Groups** from the menu bar.



	DPM
Encryption Group Name	DPM
Group Status	OK - Converged
Deployment Mode	Transparent
Failback Mode	Automatic
Key Vault Type	RSA Data Protection Manager (DPM)
REPL Support	Disabled
Primary Key Vault IP Address (IPv4 or hostname)	10.38.145.22
Primary Key Vault Connection Status	Connected
Backup Key Vault IP Address (IPv4 or hostname)	None
Backup Key Vault Connection Status	Key Vault Not Configured
High Availability Mode	(Not Applicable)
User Authentication	(Not Applicable)
Certificate Type	(Not Applicable)
Vendor Name	(Not Applicable)

If you specify a key vault IP address above, then you must enter a key vault certificate below.
If a key vault address is not specified above, then entries below are ignored.

Primary Key Vault Certificate

```
Version: V3
Subject: CN=RSA Key Manager Appliance Demo Root CA - 2012-08-20 15:12:35
Signature Algorithm: SHA1withRSA, OID = 1.2.840.113549.1.1.5
Key:  Sun RSA public key, 2048 bits
```

Backup Key Vault Certificate

<None>

FIGURE 16 **Encryption Group Properties with Key Vault Certificate**

2. Select **Load from File** and browse to the location on your client PC that contains the downloaded CA certificate in .pem format.

Encryption preparation

Before you use the encryption setup wizard for the first time, you should have a detailed configuration plan in place and available for reference. The encryption setup wizard assumes the following:

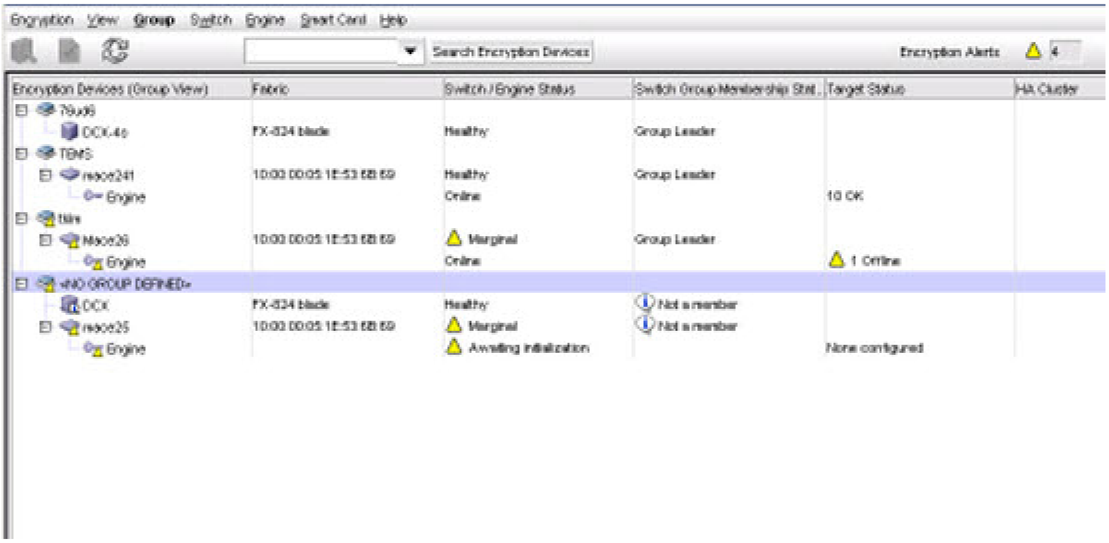
- You have a plan in place to organize encryption devices into encryption groups.
- If you want redundancy and high availability in your implementation, you have a plan to create high availability (HA) clusters of two encryption switches or blades to provide failover support.
- All switches in the planned encryption group are interconnected on an I/O synch LAN.
- The management ports on all encryption switches and 8-slot Backbone Chassis CPs that have encryption blades installed, have a LAN connection to the SAN management program and are available for discovery.
- A supported key management appliance is connected on the same LAN as the encryption switches, 8-slot Backbone Chassis CPs, and the SAN Management program.
- An external host is available on the LAN to facilitate certificate exchange.
- Switch KAC certificates have been signed by a CA and stored in a known location.
- Key management system (key vault) certificates have been obtained and stored in a known location.

Creating an encryption group

The following steps describe how to start and run the encryption setup wizard and create a new encryption group.

NOTE
When a new encryption group is created, any existing tape pools in the switch are removed.

1. Select **Configure** > **Encryption** from the menu task bar to display the **Encryption Center** dialog box (Figure 17).



Encryption Devices (Group View)	Fabric	Switch / Engine Status	Switch Group Membership Stat...	Target Status	HA Cluster
79ud6					
DCX-4s	FX-824 blade	Healthy	Group Leader		
TEMS					
mace241	10:00:00:05:1E:53:6B:69	Healthy	Group Leader		
Engine		Online		10 OK	
tkis					
Mace26	10:00:00:05:1E:53:6B:69	⚠ Marginal	Group Leader		
Engine		Online		⚠ 1 Offline	
<NO GROUP DEFINED>					
DCX	FX-824 blade	Healthy	ⓘ Not a member		
mace25	10:00:00:05:1E:53:6B:69	⚠ Marginal	ⓘ Not a member		
Engine		⚠ Awaiting initialization		None configured	

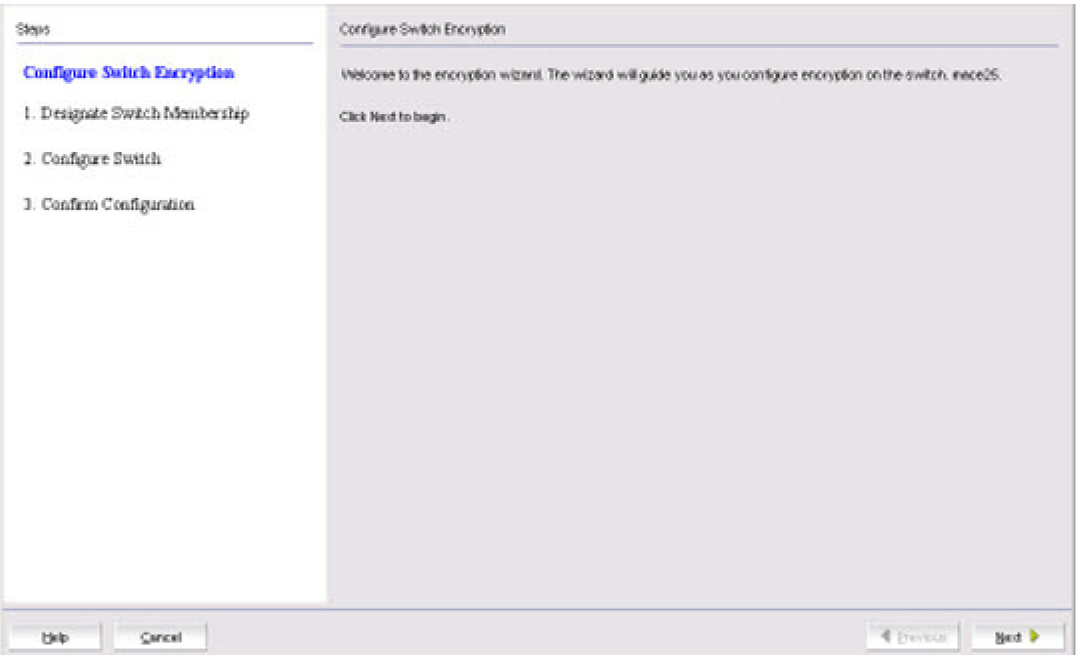
FIGURE 17 **Encryption Center dialog box - No group defined**

2. Select a switch from the **<NO GROUP DEFINED>** encryption group. (The switch must not be assigned to an encryption group.)
3. Select **Encryption > Create/Add to Group**, from the menu task bar.

The **Configure Switch Encryption** wizard welcome screen displays (Figure 18). The wizard enables you to create a new encryption group, or add an encryption switch to an existing encryption group. The wizard also enables you to configure switch encryption.

Click **Next** on each screen to advance to the next step in the wizard. Steps might vary slightly depending on the key vault type selected, but the basic wizard steps are as follows.

1. Designate Switch Membership.
2. Create a new encryption group or add a switch to an existing encryption group.
3. Select the key vault.
4. Specify the public key filename.
5. Select Security Settings.
6. Confirm the configuration.
7. Configuration Status.
8. Read Instructions.


Steps
Configure Switch Encryption
1. Designate Switch Membership
2. Configure Switch
3. Confirm Configuration
Configure Switch Encryption
Help
Cancel
Previous
Next

Welcome to the encryption wizard. The wizard will guide you as you configure encryption on the switch, mace25.

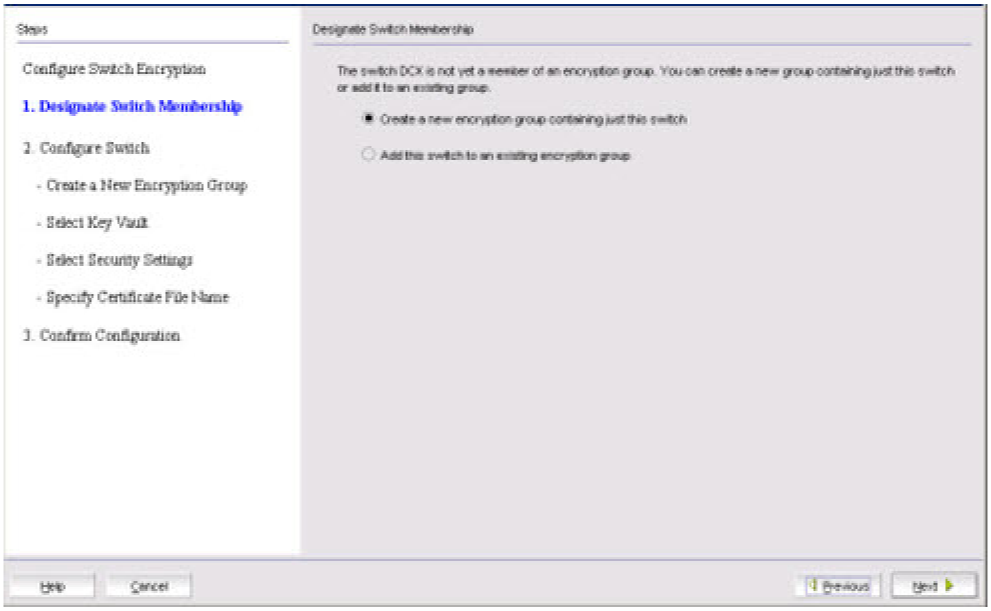
Click Next to begin.

FIGURE 18 Configure Switch Encryption wizard - welcome screen

4. From the **Configure Switch Encryption** welcome screen, click **Next** to begin.

   The **Designate Switch Membership** dialog box displays (Figure 19). The dialog box contains the following options:

   - **Create a new encryption group containing just the switch**: Creates an encryption group for the selected switch
   - **Add this switch to an existing encryption group**: Adds the selected switch to an encryption group that already exists


Steps
Configure Switch Encryption
1. Designate Switch Membership
2. Configure Switch
- Create a New Encryption Group
- Select Key Vault
- Select Security Settings
- Specify Certificate File Name
3. Confirm Configuration
Designate Switch Membership
Help
Cancel
Previous
Next

The switch DCX is not yet a member of an encryption group. You can create a new group containing just this switch or add it to an existing group.

● Create a new encryption group containing just this switch

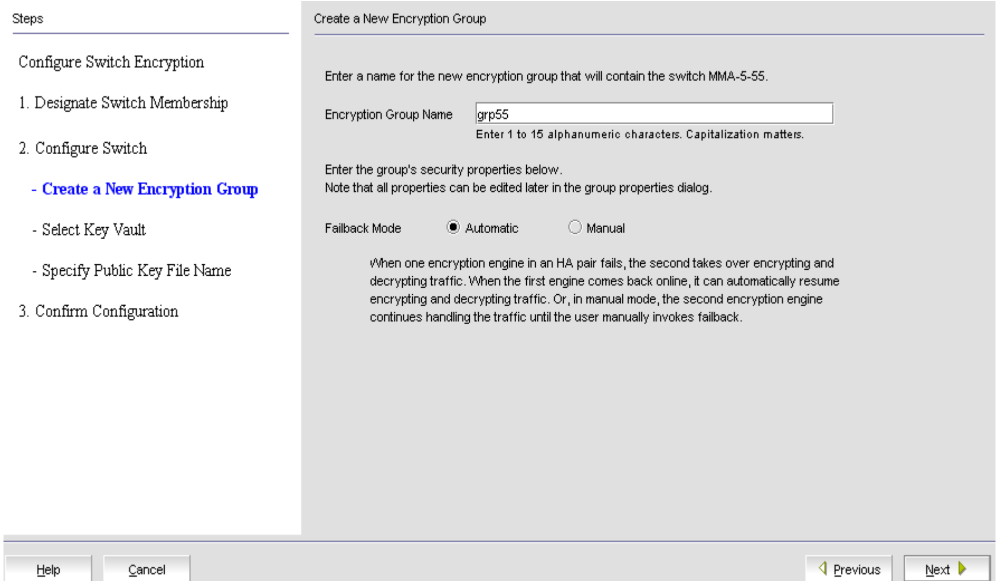
○ Add this switch to an existing encryption group

FIGURE 19 Designate Switch Membership dialog box

5. For this procedure, verify that **Create a new encryption group containing just this switch** is selected, then click **Next.**

   NOTE
   If you are adding a switch to an encryption, refer to "Adding a switch to an encryption group" on page 46.

   The **Create a New Encryption Group** dialog box displays (Figure 20).



Steps

Configure Switch Encryption

1. Designate Switch Membership

2. Configure Switch

- Create a New Encryption Group

- Select Key Vault

- Specify Public Key File Name

3. Confirm Configuration

Create a New Encryption Group

Enter a name for the new encryption group that will contain the switch MMA-5-55.

Encryption Group Name: grp55

Enter 1 to 15 alphanumeric characters. Capitalization matters.

Enter the group's security properties below.
Note that all properties can be edited later in the group properties dialog.

Failback Mode ● Automatic ○ Manual

When one encryption engine in an HA pair fails, the second takes over encrypting and decrypting traffic. When the first engine comes back online, it can automatically resume encrypting and decrypting traffic. Or, in manual mode, the second encryption engine continues handling the traffic until the user manually invokes failback.

FIGURE 20 Create a New Encryption Group dialog box

The dialog box contains the following information:

- **Encryption Group Name** text box: Encryption group names can have up to 15 characters. Letters, digits, and underscores are allowed. The group name is case-sensitive.
- **Failback mode**: Selects whether or not storage targets should be automatically transferred back to an encryption engine that comes online after being unavailable. Options are **Automatic** or **Manual**.

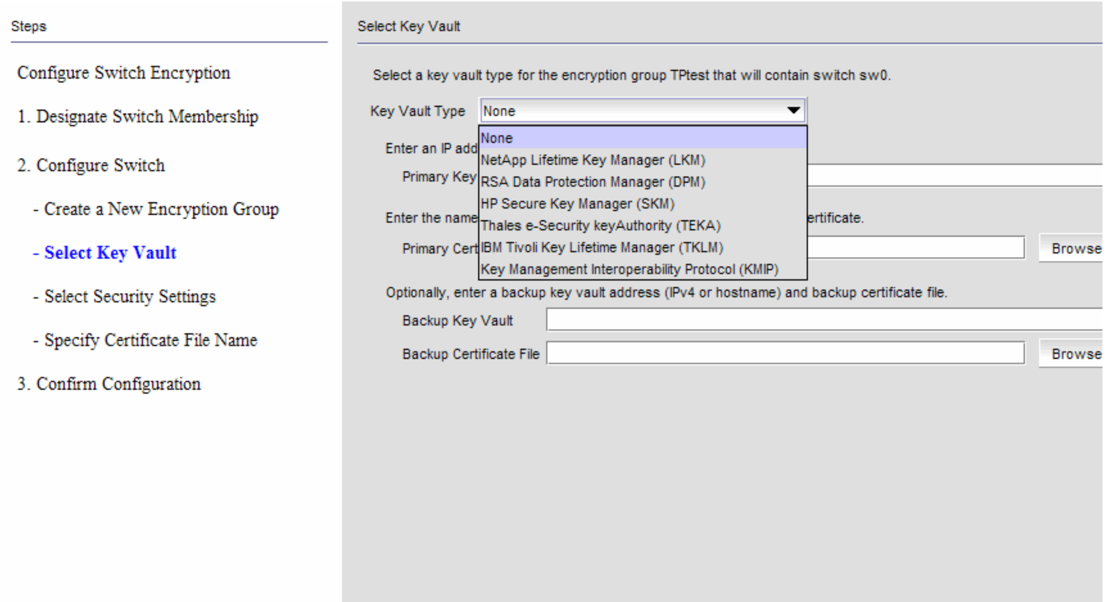
NOTE
When one encryption engine in the HA cluster fails, the second encryption engine in the HA cluster takes over the encryption and decryption of traffic to all encryption targets in the first encryption engine (failover). When the first encryption engine comes back online, the encryption group's failback setting (auto or manual) determines whether the first encryption engine automatically resumes encrypting and decrypting traffic to its encryption targets. In manual mode, the second encryption engine continues to handle the traffic until you manually invoke failback by way of the **Encryption Targets** dialog box.

6. Enter an **Encryption Group Name** for the encryption group and select **Automatic** as the Failback mode.

   If the name for the encryption group already exists, a pop-up warning message displays. Although unique group names avoid confusion while managing multiple groups, you are not prevented from using duplicate group names. Click **Yes** to use the same name for the new encryption group, or click **No** to enter another name.

7. Click **Next**.

   The **Select Key Vault.** dialog box displays (Figure 21).



Steps

Configure Switch Encryption

1. Designate Switch Membership

2. Configure Switch

- Create a New Encryption Group
- **Select Key Vault**
- Select Security Settings
- Specify Certificate File Name

3. Confirm Configuration

Select Key Vault

Select a key vault type for the encryption group TPtest that will contain switch sw0.

Key Vault Type: None

Enter an IP add

Primary Key

Enter the name ertificate.

Primary Cert

Optionally, enter a backup key vault address (IPv4 or hostname) and backup certificate file.

Backup Key Vault

Backup Certificate File

None
NetApp Lifetime Key Manager (LKM)
RSA Data Protection Manager (DPM)
HP Secure Key Manager (SKM)
Thales e-Security keyAuthority (TEKA)
IBM Tivoli Key Lifetime Manager (TKLM)
Key Management Interoperability Protocol (KMIP)
Browse
Browse

FIGURE 21 **Select Key Vault dialog box**

Using this dialog box, you can select a key vault for the encryption group that contains the selected switch. Prior to selecting your Key Vault Type, the selection is shown as None. The dialog box contains the following information:

- Key Vault Type:

  If an encryption group contains mixed firmware nodes, the Encryption Group Properties Key Vault Type name is based on the firmware version of the group leader.

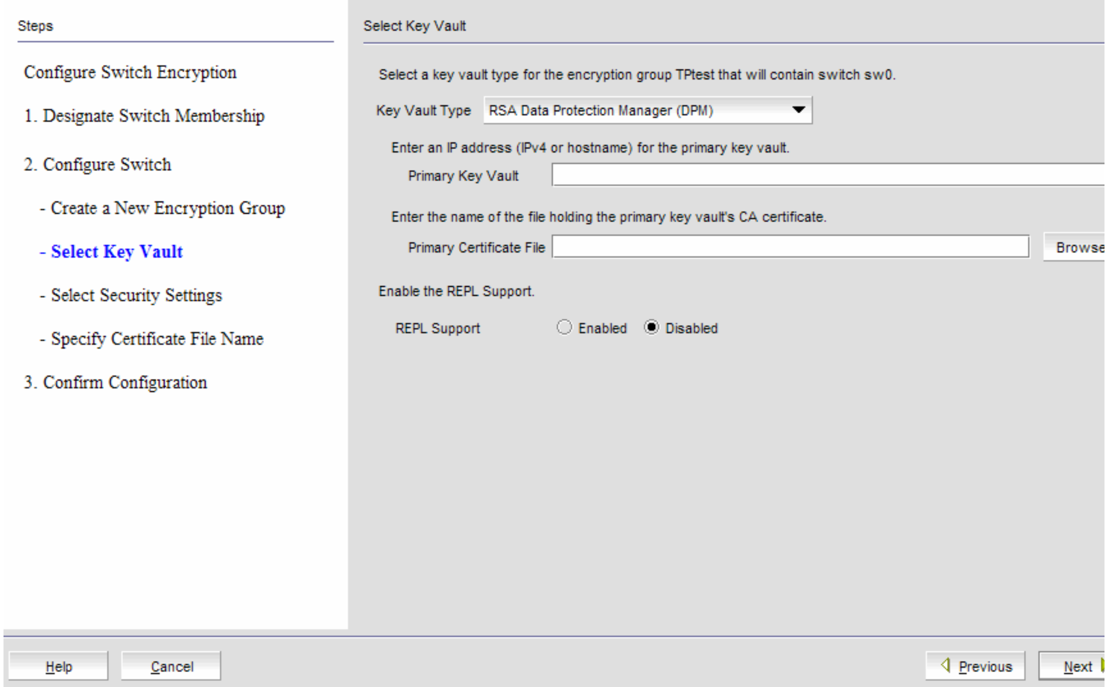
- Options are:
  - NetApp Link Key Manager (LKM)
  - RSA Data Protection Manager (DPM): If an encryption group contains mixed firmware nodes, the Encryption Group Properties Key Vault Type name is based on the firmware version of the group leader. For example, If a switch is running Fabric OS 7.1.0 or later, the Key Vault Type is displayed as “RSA Data Protection Manager (DPM).”If a switch is running a Fabric OS version prior to v7.1.0, Key Vault Type is displayed as “RSA Key Manager (RKM)”.
  - HP Secure Key Manager (SKM)
  - Thales e-Security keyAuthority (TEKA)
  - Tivoli Key Lifecycle Manager (TKLM)
  - Key Management Interoperability Protocol (KMIP)

8. Select RSA Data Protection Manager (DPM) as the Key Vault Type.

Configuring key vault settings for RSA Data Protection Manager (DPM)

The following procedure assumes you have already configured the initial steps in the **Configure Switch Encryption** wizard. If you have not already done so, go to “Creating an encryption group” on page 35.

Figure 22 shows the key vault selection dialog box for DPM.


Steps
Configure Switch Encryption
1. Designate Switch Membership
2. Configure Switch
- Create a New Encryption Group
- Select Key Vault
- Select Security Settings
- Specify Certificate File Name
3. Confirm Configuration
Select Key Vault
Browse
Help
Cancel
Previous
Next

Select a key vault type for the encryption group TPtest that will contain switch sw0.

Key Vault Type: RSA Data Protection Manager (DPM)

Enter an IP address (IPv4 or hostname) for the primary key vault.

Primary Key Vault:

Enter the name of the file holding the primary key vault's CA certificate.

Primary Certificate File:

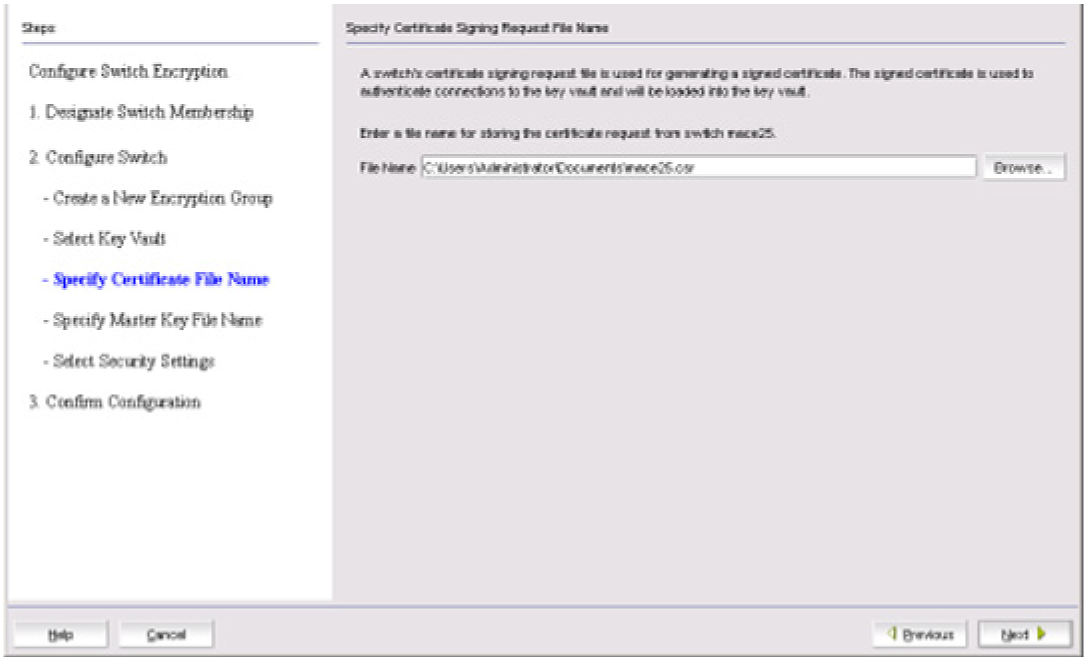
Enable the REPL Support.

REPL Support ○ Enabled ● Disabled

FIGURE 22 Select Key Vault dialog box for DPM

1. Enter the IPv4 IP address or host name for the primary key vault. If you are clustering DPM appliances for high availability, IP load balancers are used to direct traffic to the appliances. Use the IP address of the load balancer.
2. Enter the name of the file that holds the Primary Key Vault’s CA Key Certificate file, or browse to the desired location. This file can be generated from the key vault’s administrative console.
3. If you are implementing encryption on data replication LUNs used by the EMC Symmetrix Remote Data Facility (SRDF), you must select **Enabled** for **REPL Support**.
4. Click **Next**.

   The **Specify Certificate Signing Request File Name** dialog box displays (Figure 23).


Steps
Configure Switch Encryption
1. Designate Switch Membership
2. Configure Switch
- Create a New Encryption Group
- Select Key Vault
- Specify Certificate File Name
- Specify Master Key File Name
- Select Security Settings
3. Confirm Configuration
Browse...
Help
Cancel
Previous
Next

Specify Certificate Signing Request File Name

A switch's certificate signing request file is used for generating a signed certificate. The signed certificate is used to authenticate connections to the key vault and will be loaded into the key vault.

Enter a file name for storing the certificate request from switch mace25.

File Name C:\Users\Administrator\Documents\mace25.csr

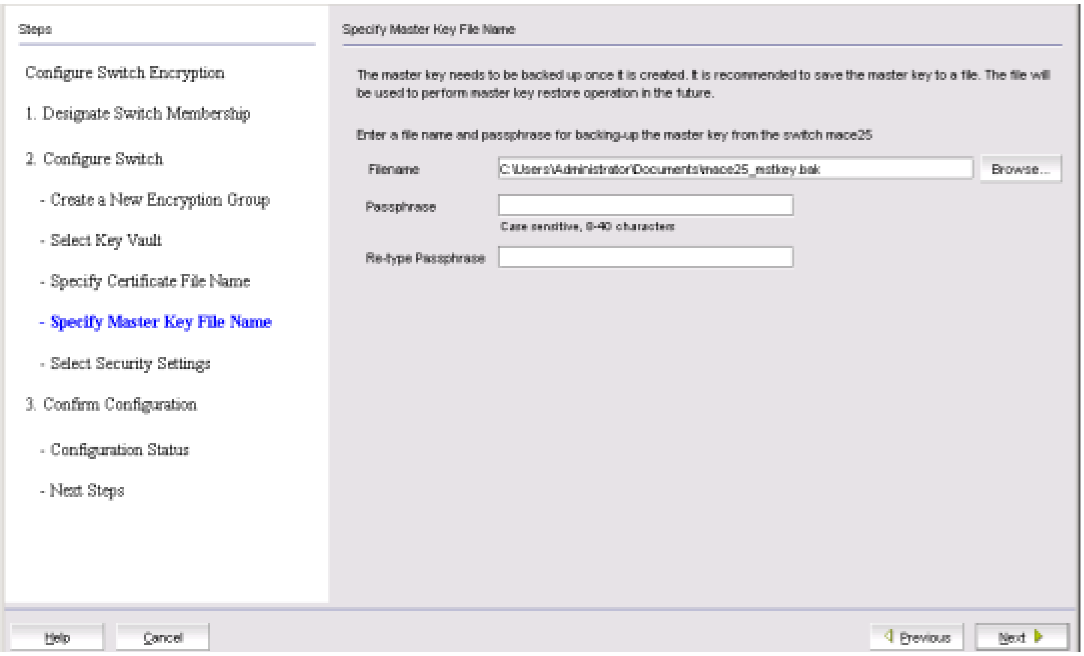
FIGURE 23 Specify Certificate Signing Request File Name dialog box

5. Enter the filename in which you want to store the certificate information, or browse to the file location.

   The certificate stored in this file is the switch's Switch Certificate Signing file. You will need to know this path and file name to install the switch's Switch Certificate Signing file on the key management appliance.

6. Click **Next**.

   The **Specify Master Key File Name dialog box** displays (Figure 24).


Steps
Configure Switch Encryption
1. Designate Switch Membership
2. Configure Switch
- Create a New Encryption Group
- Select Key Vault
- Specify Certificate File Name
- Specify Master Key File Name
- Select Security Settings
3. Confirm Configuration
- Configuration Status
- Next Steps
Help
Cancel
Previous
Next
Browse...

Specify Master Key File Name

The master key needs to be backed up once it is created. It is recommended to save the master key to a file. The file will be used to perform master key restore operation in the future.

Enter a file name and passphrase for backing-up the master key from the switch mace25

Filename: C:\Users\Administrator\Documents\mace25_mstkey.bak

Passphrase:

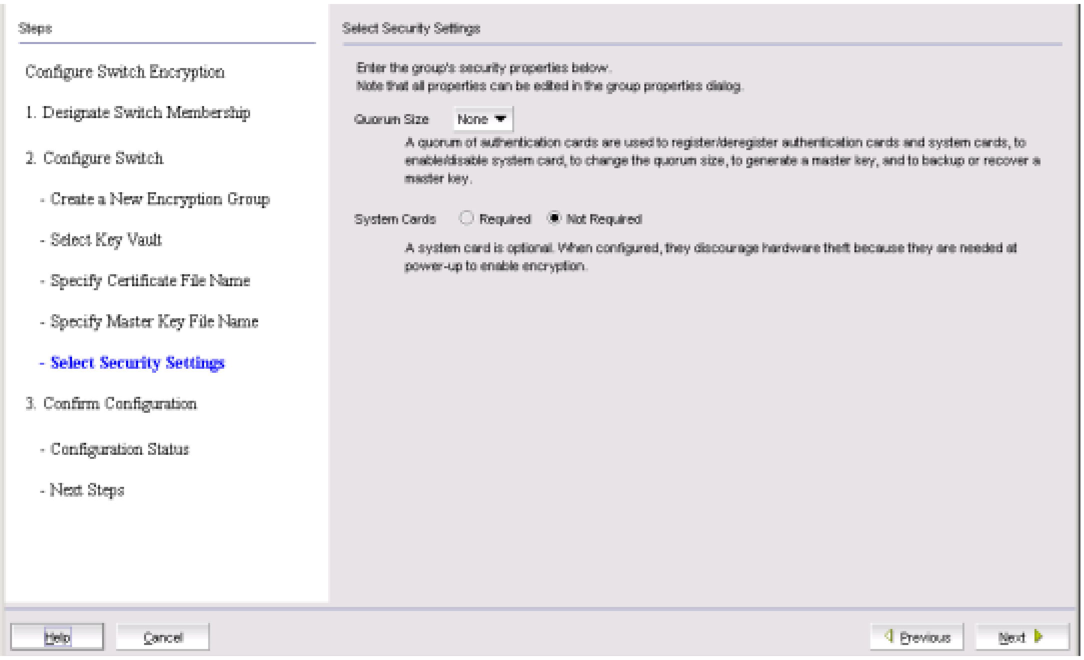
Case sensitive, 8-40 characters

Re-type Passphrase:

FIGURE 24 Specify Master Key File Name dialog box

7. Enter the location of the file where you want to store back up master key information, or browse to the desired location.
8. Enter the passphrase, which is required for restoring the master key. The passphrase can be between eight and 40 characters, and any character is allowed.
9. Re-enter the passphrase for verification, then click **Next**.

   The **Select Security Settings** dialog box displays (Figure 25).


Steps

Configure Switch Encryption

1. Designate Switch Membership

2. Configure Switch

- Create a New Encryption Group
- Select Key Vault
- Specify Certificate File Name
- Specify Master Key File Name
- **Select Security Settings**

3. Confirm Configuration

- Configuration Status
- Next Steps

Select Security Settings

Enter the group's security properties below.
Note that all properties can be edited in the group properties dialog.

Quorum Size: None

A quorum of authentication cards are used to register/deregister authentication cards and system cards, to enable/disable system card, to change the quorum size, to generate a master key, and to backup or recover a master key.

System Cards: ○ Required ● Not Required

A system card is optional. When configured, they discourage hardware theft because they are needed at power-up to enable encryption.

FIGURE 25 Select Security Settings dialog box

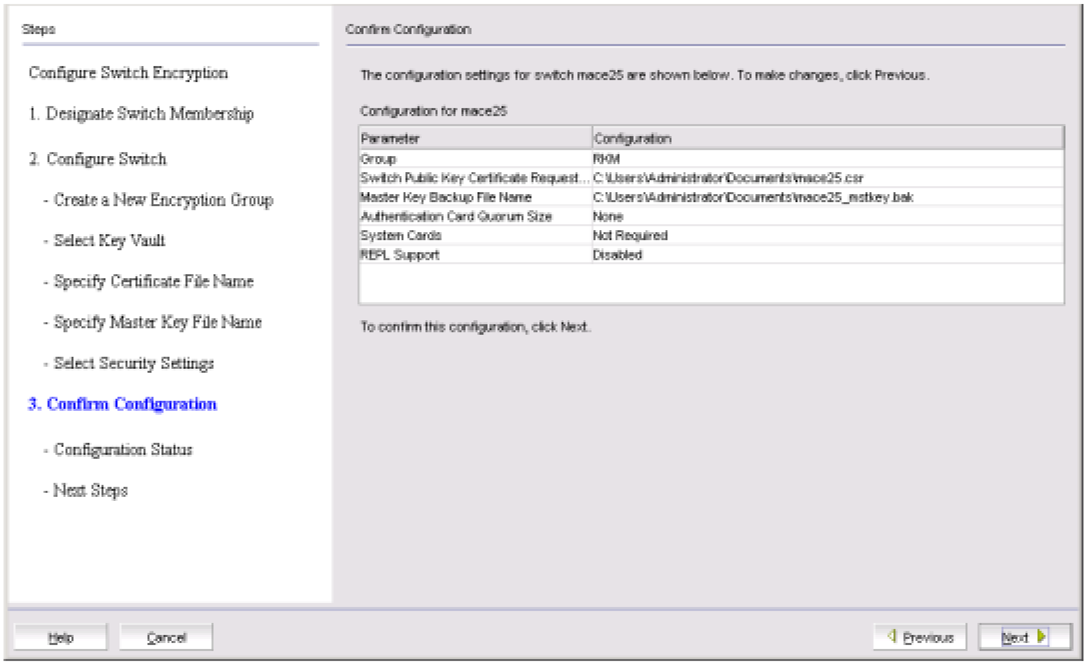
10. Set quorum size and system card requirements.

    The quorum size is the minimum number of cards necessary to enable the card holders to perform the security sensitive operations listed above. The maximum quorum size is five cards. The actual number of authentication cards registered is always more than the quorum size, so if you set the quorum size to five, for example, you will need to register at least six cards in the subsequent steps.

    Setting quorum size to a value greater than zero and/or setting system cards to **Required** launches additional wizard dialog boxes.

11. Click **Next**.

    The **Confirm Configuration** dialog box displays (Figure 26). Confirm the encryption group name and switch public key certificate file name you specified are correct, then click **Next**.



Steps

Configure Switch Encryption

1. Designate Switch Membership
2. Configure Switch
   - Create a New Encryption Group
   - Select Key Vault
   - Specify Certificate File Name
   - Specify Master Key File Name
   - Select Security Settings
3. Confirm Configuration
   - Configuration Status
   - Next Steps

Confirm Configuration

The configuration settings for switch mace25 are shown below. To make changes, click Previous.

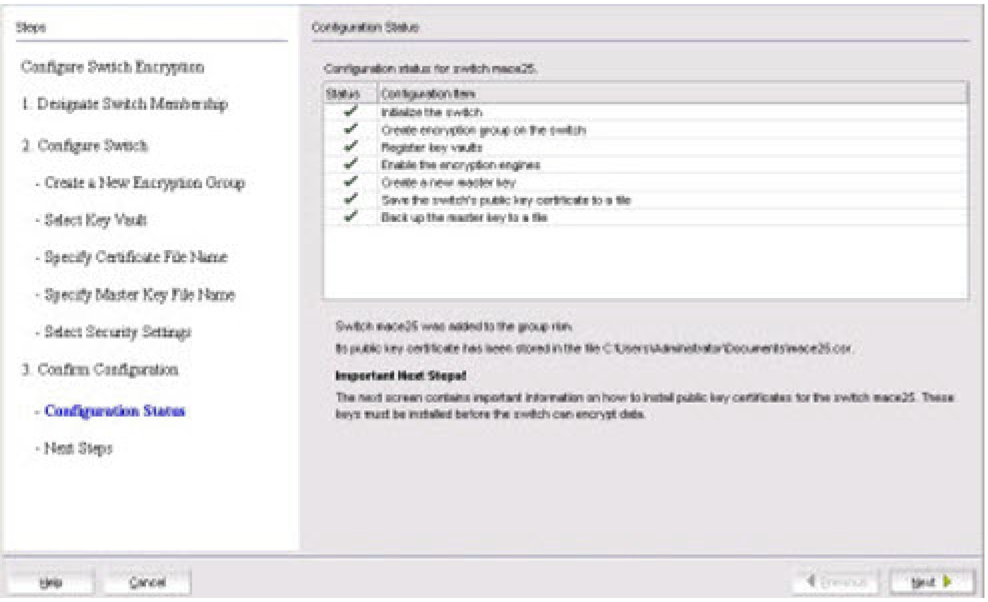
Configuration for mace25

Parameter	Configuration
Group	RKM
Switch Public Key Certificate Request...	C:\Users\Administrator\Documents\mace25.csr
Master Key Backup File Name	C:\Users\Administrator\Documents\mace25_mstkey.bak
Authentication Card Quorum Size	None
System Cards	Not Required
REPL Support	Disabled

To confirm this configuration, click Next.

FIGURE 26 Confirm Configuration dialog box

The **Configuration Status** dialog box displays (Figure 27).



Steps

Configure Switch Encryption

1. Designate Switch Membership
2. Configure Switch
   - Create a New Encryption Group
   - Select Key Vault
   - Specify Certificate File Name
   - Specify Master Key File Name
   - Select Security Settings
3. Confirm Configuration
   - Configuration Status
   - Next Steps

Configuration Status

Configuration status for switch mace25.

Status	Configuration Item
✓	Initialize the switch
✓	Create encryption group on the switch
✓	Register key vaults
✓	Enable the encryption engines
✓	Create a new master key
✓	Save the switch's public key certificate to a file
✓	Back up the master key to a file

Switch mace25 was added to the group rkm.

Its public key certificate has been stored in the file C:\Users\Administrator\Documents\mace25.csr.

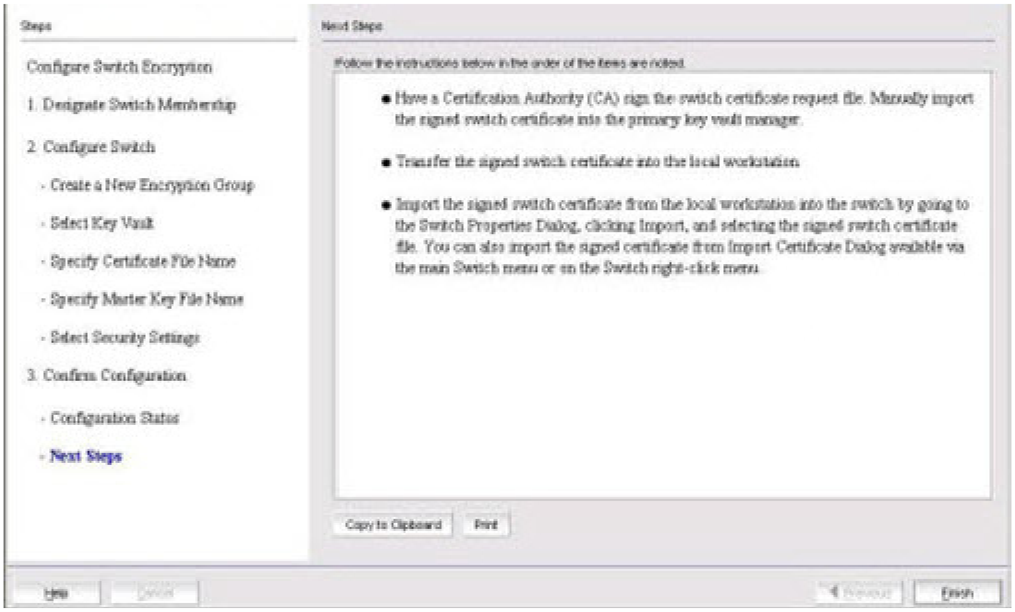
Important Next Steps!

The next screen contains important information on how to install public key certificates for the switch mace25. These keys must be installed before the switch can encrypt data.

FIGURE 27 Configuration Status dialog box

12. Review the post-configuration instructions, which you can copy to a clipboard or print for later, then click **Next**.

    The **Next Steps** dialog box displays (Figure 28). Instructions for installing public key certificates for the encryption switch are displayed.


Steps
Configure Switch Encryption
1. Designate Switch Membership
2. Configure Switch
- Create a New Encryption Group
- Select Key Vault
- Specify Certificate File Name
- Specify Master Key File Name
- Select Security Settings
3. Confirm Configuration
- Configuration Status
- Next Steps
Next Steps
Copy to Clipboard
Print
Help
Cancel
Previous
Finish

Follow the instructions below in the order of the items are noted.

- Have a Certification Authority (CA) sign the switch certificate request file. Manually import the signed switch certificate into the primary key vault manager.
- Transfer the signed switch certificate into the local workstation.
- Import the signed switch certificate from the local workstation into the switch by going to the Switch Properties Dialog, clicking Import, and selecting the signed switch certificate file. You can also import the signed certificate from Import Certificate Dialog available via the main Switch menu or on the Switch right-click menu.

FIGURE 28 Next Steps dialog box

13. Review the post-configuration instructions, which you can copy to a clipboard or print for later, then click **Finish** to exit the wizard.

Understanding configuration status results

After configuration of the encryption group is completed, BNA sends API commands to verify the switch configuration. The CLI commands are detailed in the encryption administrator's guide for your key vault management system.

1. Initialize the switch. If the switch is not already in the initiated state, BNA performs the **cryptocfg --initnode** command.
2. Create an encryption group on the switch. BNA creates a new group using the **cryptocfg --create -encgroup** command, and sets the key vault type using the **cryptocfg --set -keyvault** command.
3. Register the key vault. BNA registers the key vault using the **cryptocfg --reg keyvault** command.
4. Enable the encryption engines. BNA initializes an encryption switch using the **cryptocfg --initEE [<slotnumber>] and cryptocfg --regEE [<slotnumber>]** commands.
5. Create a new master key. (Opaque key vaults only). BNA checks for a new master key. New master keys are generated from the **Security** tab located in the **Encryption Group Properties** dialog box.
6. Save the switch's public key certificate to a file. BNA saves the KAC certificate in the specified file.
7. Back up the master key to a file. (Opaque key vaults only). BNA saves the master key in the specified file.

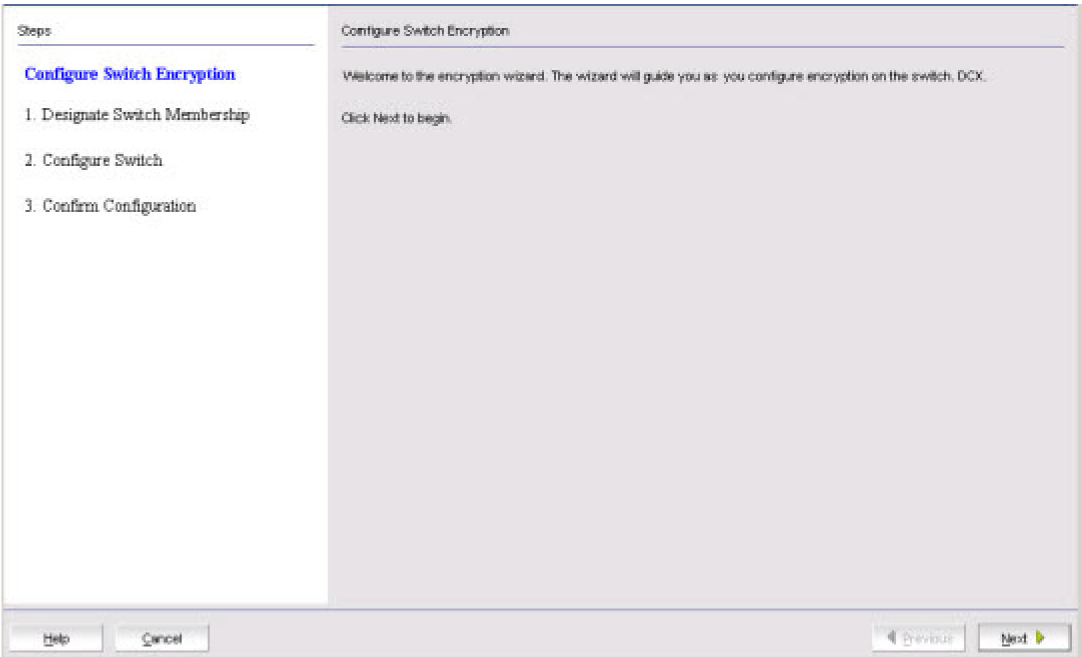
Adding a switch to an encryption group

The setup wizard allows you to either create a new encryption group, or add an encryption switch to an existing encryption group. Use the following procedure to add a switch to an encryption group:

1. Select **Configure > Encryption** from the menu task bar to display the **Encryption Center** dialog box (Refer to Figure 6 on page 14).
2. Select a switch to add from the **Encryption Center Devices** table, then select **Switch > Create/Add to Group** from the menu task bar.

   NOTE
   The switch must not already be in an encryption group.

   The **Configure Switch Encryption** wizard welcome screen displays (Figure 29).


Steps
Configure Switch Encryption
1. Designate Switch Membership
2. Configure Switch
3. Confirm Configuration
Configure Switch Encryption
Help
Cancel
Previous
Next

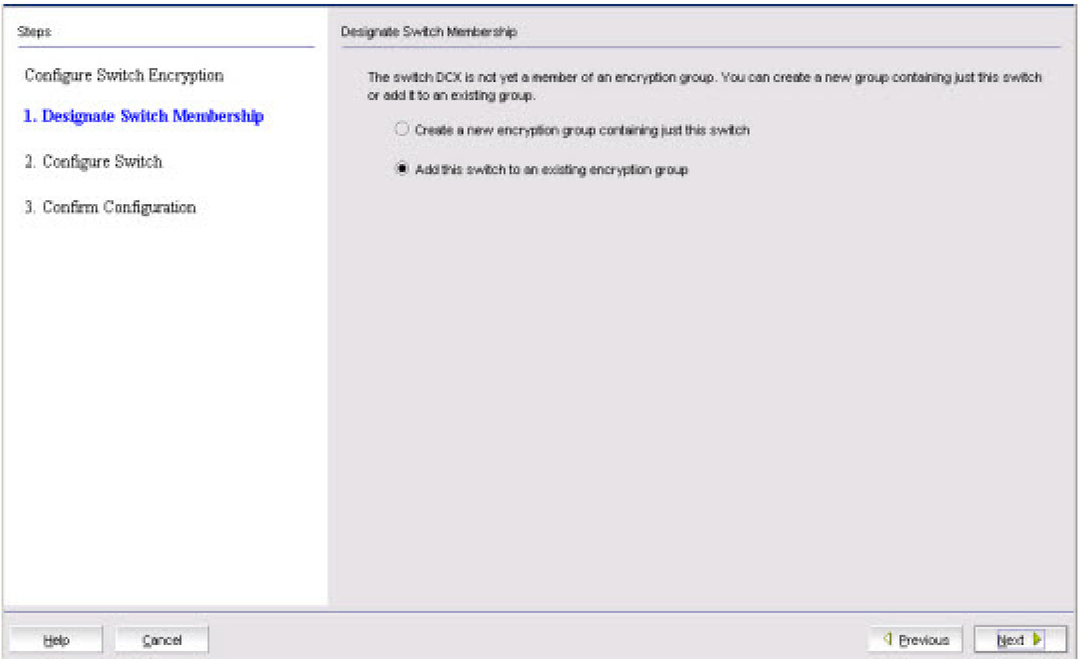
Welcome to the encryption wizard. The wizard will guide you as you configure encryption on the switch, DCX.

Click Next to begin.

FIGURE 29 Configure Switch Encryption wizard - welcome screen

3. Click **Next**.

   The **Designate Switch Membership** dialog box displays (Figure 30).



Configure Switch Encryption

1. Designate Switch Membership

2. Configure Switch

3. Confirm Configuration

The switch DCX is not yet a member of an encryption group. You can create a new group containing just this switch or add it to an existing group.

○ Create a new encryption group containing just this switch

● Add this switch to an existing encryption group

FIGURE 30 **Designate Switch Membership dialog box**

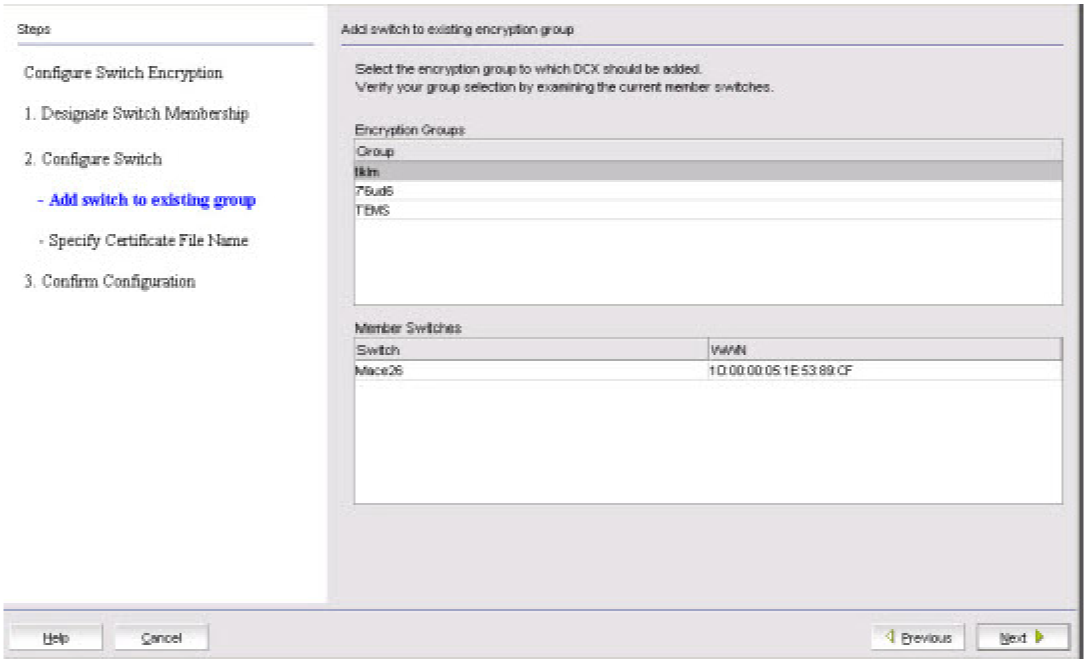
4. For this procedure, select **Add this switch to an existing encryption group**, then click **Next**.

   The **Add Switch to Existing Encryption Group** dialog box displays (Figure 31).

   The dialog box contains the following information:

   - **Encryption Groups** table: Enables you to select an encryption group in which to add a switch.
   - **Member Switches** table: Lists the switches in the selected encryption group.

NOTE
If you are creating a new encryption group, refer to "Creating an encryption group" on page 35.


Steps
Configure Switch Encryption
1. Designate Switch Membership
2. Configure Switch
- Add switch to existing group
- Specify Certificate File Name
3. Confirm Configuration
Help
Cancel
Previous
Next

Add switch to existing encryption group

Select the encryption group to which DCX should be added.
Verify your group selection by examining the current member switches.

Encryption Groups

Group
tkim
75ud6
TEMS

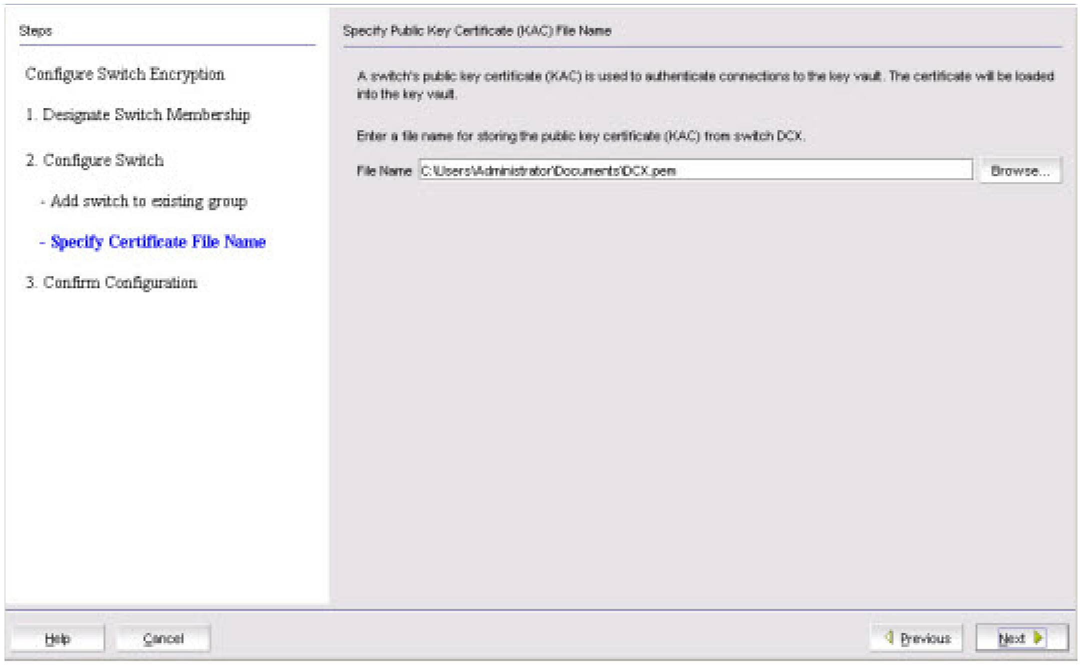
Member Switches

Switch	WWN
Mace26	10:00:00:05:1E:53:89:CF

FIGURE 31 Add Switch to Existing Encryption Group dialog box

5. Select the group in which to add the switch, then click **Next**.

   The **Specify Public Key Certificate (KAC) File Name** dialog box displays (Figure 32).


Steps
Configure Switch Encryption
1. Designate Switch Membership
2. Configure Switch
- Add switch to existing group
- Specify Certificate File Name
3. Confirm Configuration
Browse...
Help
Cancel
Previous
Next

Specify Public Key Certificate (KAC) File Name

A switch's public key certificate (KAC) is used to authenticate connections to the key vault. The certificate will be loaded into the key vault.

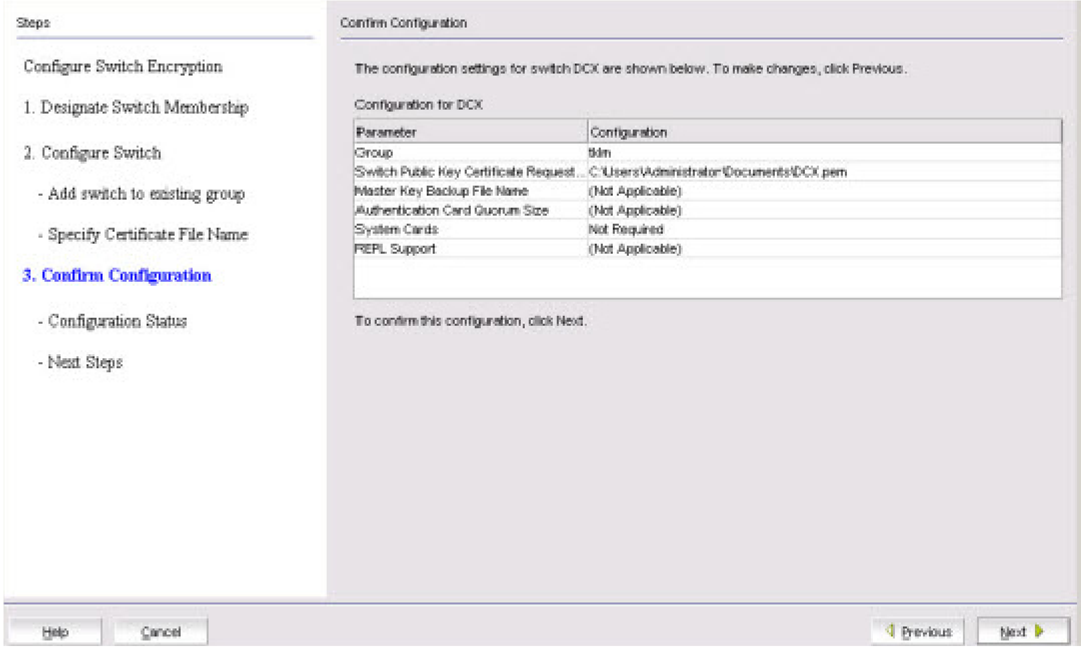
Enter a file name for storing the public key certificate (KAC) from switch DCX.

File Name C:\Users\Administrator\Documents\DCX.pem

FIGURE 32 Specify Public Key Certificate (KAC) File Name dialog box

6. Enter the location where you want to store the public key certificate that is used to authenticate connections to the key vault, or browse to the desired location, then click **Next**.

   The **Confirm Configuration** dialog box displays (Figure 33). Confirm the encryption group name and switch public key certificate file name you specified are correct, then click **Next**.



Steps
Configure Switch Encryption
1. Designate Switch Membership
2. Configure Switch
- Add switch to existing group
- Specify Certificate File Name
3. Confirm Configuration
- Configuration Status
- Next Steps
Help
Cancel
Previous
Next

Confirm Configuration

The configuration settings for switch DCX are shown below. To make changes, click Previous.

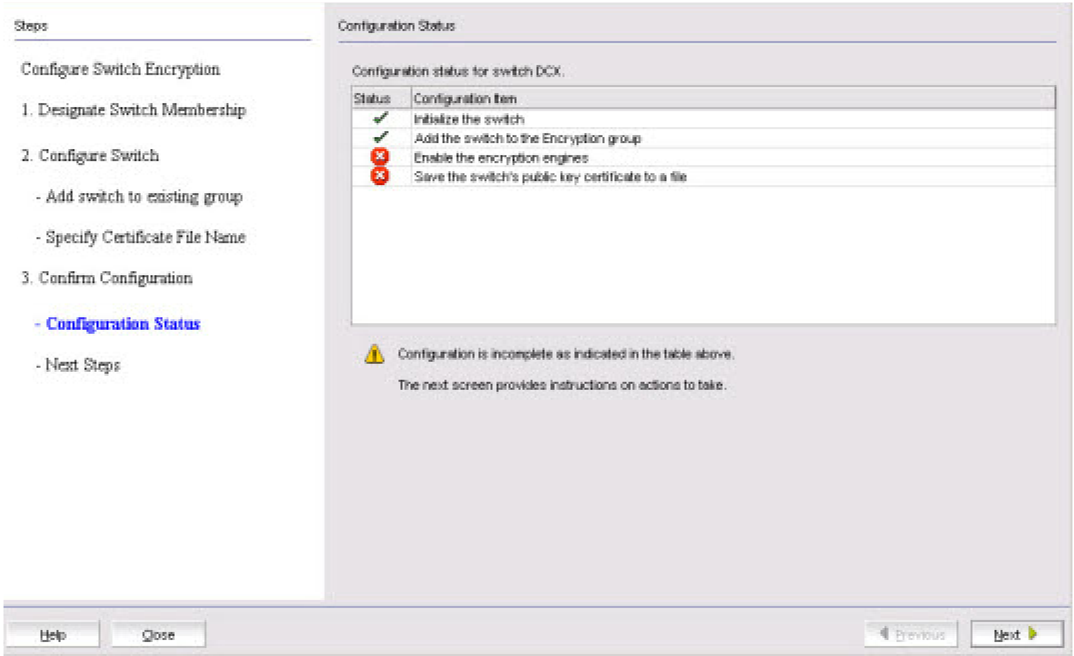
Configuration for DCX

Parameter	Configuration
Group	tldm
Switch Public Key Certificate Request...	C:\Users\Administrator\Documents\DCX.pem
Master Key Backup File Name	(Not Applicable)
Authentication Card Quorum Size	(Not Applicable)
System Cards	Not Required
REPL Support	(Not Applicable)

To confirm this configuration, click Next.

FIGURE 33 Confirm Configuration dialog box

The **Configuration Status** dialog box displays (Figure 34).



Steps
Configure Switch Encryption
1. Designate Switch Membership
2. Configure Switch
- Add switch to existing group
- Specify Certificate File Name
3. Confirm Configuration
- Configuration Status
- Next Steps
Help
Close
Previous
Next

Configuration Status

Configuration status for switch DCX.

Status	Configuration Item
✓	Initialize the switch
✓	Add the switch to the Encryption group
☒	Enable the encryption engines
☒	Save the switch's public key certificate to a file

⚠ Configuration is incomplete as indicated in the table above.

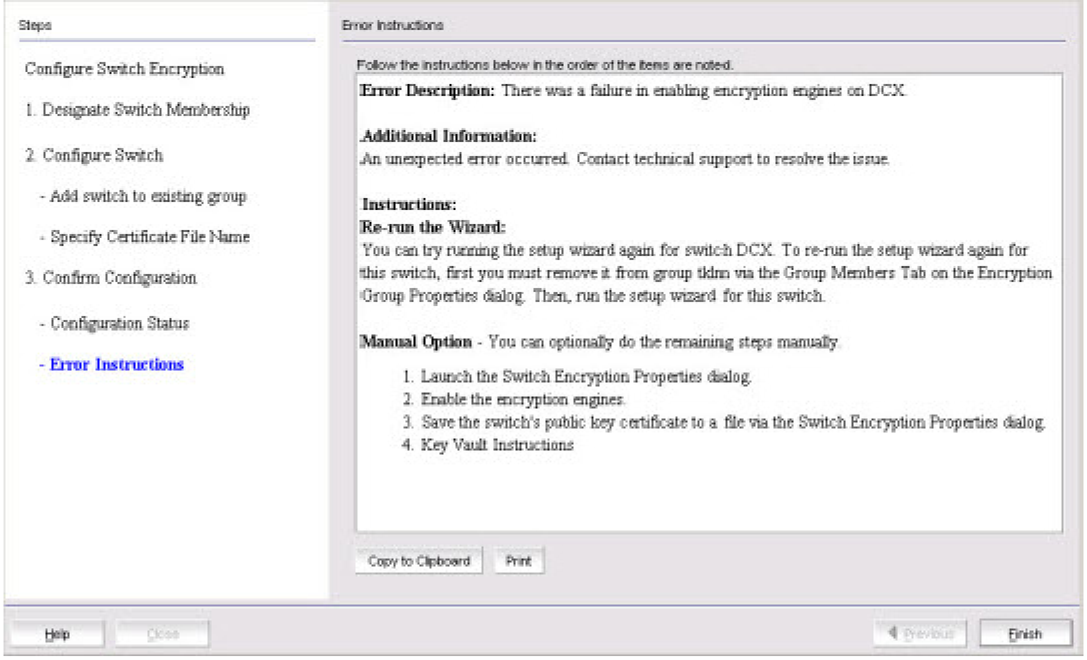
The next screen provides instructions on actions to take.

FIGURE 34 Configuration Status dialog box

All configuration items have green check marks if the configuration is successful. A red stop sign indicates a failed step. A message displays below the table, indicating the encryption switch was added to the group you named, and the public key certificate is stored in the location you specified.

7. Review important messages, then click **Next**.

   The **Error Instructions** dialog box displays (Figure 35). Instructions for installing public key certificates for the encryption switch are displayed. These instructions are specific to the key vault type.



Follow the instructions below in the order of the items are noted.

Error Description: There was a failure in enabling encryption engines on DCX.

Additional Information:
An unexpected error occurred. Contact technical support to resolve the issue.

Instructions:
Re-run the Wizard:
You can try running the setup wizard again for switch DCX. To re-run the setup wizard again for this switch, first you must remove it from group tklm via the Group Members Tab on the Encryption Group Properties dialog. Then, run the setup wizard for this switch.

Manual Option - You can optionally do the remaining steps manually.

1. Launch the Switch Encryption Properties dialog.
2. Enable the encryption engines.
3. Save the switch's public key certificate to a file via the Switch Encryption Properties dialog.
4. Key Vault Instructions

FIGURE 35 Error Instructions dialog box

8. Review the post-configuration instructions, which you can copy to a clipboard or print for later.
9. Click **Finish** to exit the **Configure Switch Encryption** wizard.

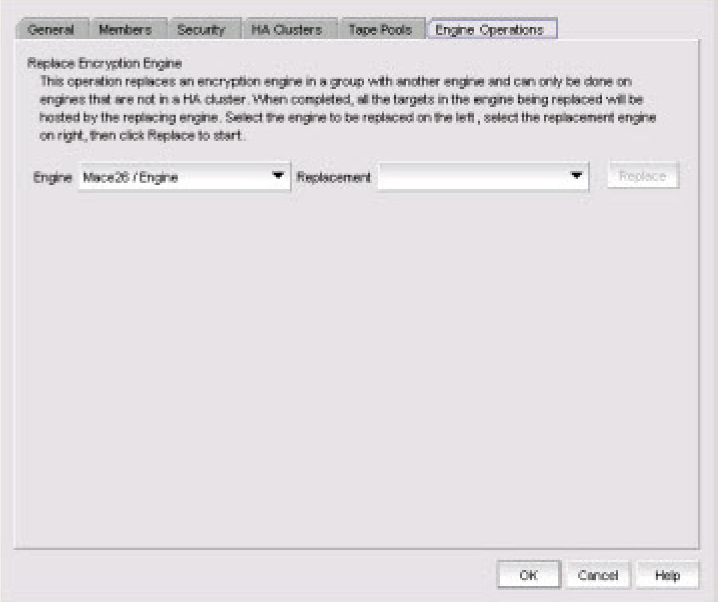
Replacing an encryption engine in an encryption group

To replace an encryption engine in an encryption group with another encryption engine within the same DEK Cluster, complete the following steps:

1. Select **Configure > Encryption** from the menu task bar to display the **Encryption Center** dialog box. (Refer to Figure 6 on page 14.)
2. Select an encryption engine from the **Encryption Center Devices** table, then select **Engine > Replace** from the menu task bar.

   The **Encryption Group Properties** dialog box displays with the **Engine Operations** tab selected (Figure 36).

   You can also display the **Engine Operations** tab by selecting an encryption group from the **Encryption Center Devices** table, selecting **Group > Properties** from the menu task bar, then selecting the **Engine Operations** tab.



Replace Encryption Engine

This operation replaces an encryption engine in a group with another engine and can only be done on engines that are not in a HA cluster. When completed, all the targets in the engine being replaced will be hosted by the replacing engine. Select the engine to be replaced on the left , select the replacement engine on right, then click Replace to start.

Engine: Mace26 / Engine

Replacement:

FIGURE 36 Engine Operations tab

3. Select the engine to replace from the **Engine** list.
4. Select the engine to use as the replacement from the **Replacement** list, then click **Replace**.

   All containers hosted by the current engine (**Engine** list) are replaced by the new engine (**Replacement** list).

High availability (HA) clusters

A high availability (HA) cluster is a group of exactly two encryption engines (EEs). One encryption engine can take over encryption and decryption tasks for the other encryption engine, if that member fails or becomes unreachable.

When creating a new HA Cluster, add one engine to create the cluster, then add the second engine. You can make multiple changes to the HA Clusters list; the changes are not applied to the switch until you click **OK**.

HA cluster configuration rules

The following rules apply when configuring an HA cluster:

- The encryption engines that are part of an HA cluster must belong to the same encryption group and be part of the same fabric.
- An HA cluster cannot span fabrics and it cannot provide failover/failback capability within a fabric transparent to host MPIO software.
- HA cluster configuration and related operations must be performed on the group leader.
- HA clusters of FS8-18 blades should not include blades in the same DCX Backbone chassis.

NOTE
In Fabric OS 6.3.0 and later, HA cluster creation is blocked when encryption engines belonging to FS8-18 blades in the same DCX Backbone chassis are specified.

- Cluster links must be configured before creating an HA cluster. Refer to the section "Configuring cluster links" on page 135 for instructions.
- It is recommended that the HA cluster configuration be completed before you configure storage devices for encryption.
- It is mandatory that the two encryption engines in the HA cluster belong to two different nodes for true redundancy. This is always true for Brocade Encryption Switches, but is not true if two FS8-18 blades in the same DCX Backbone chassis are configured in the same HA cluster.

NOTE
An IP address is required for the management port for any cluster-related operations.

Creating HA clusters

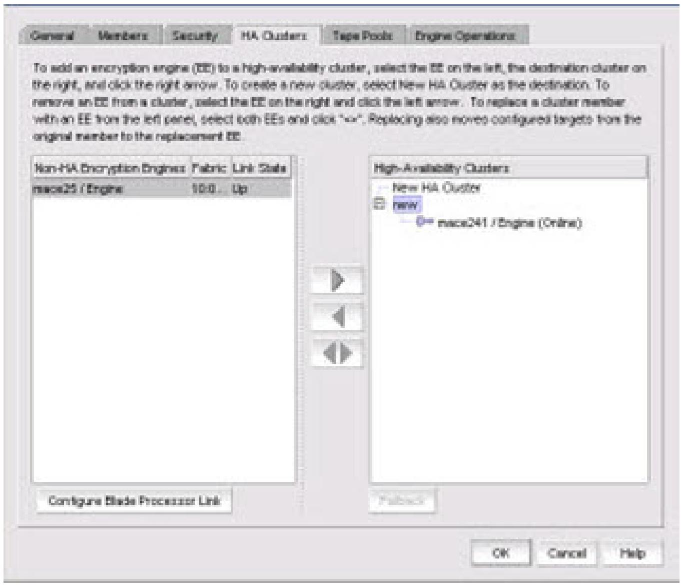
For the initial encryption node, perform the following procedure.

1. Select **Configure** > **Encryption** from the menu task bar to display the **Encryption Center** dialog box (Refer to Figure 6 on page 14).
2. Select an encryption group from the **Encryption Center Devices** table, then select **Group** > **HA Cluster** from the menu task bar.

   NOTE
   If groups are not visible in the **Encryption Center Devices** table, select **View** > **Groups** from the menu task bar.

   The **Encryption Group Properties** dialog box displays, with the **HA Clusters** tab selected (Figure 37).
3. Select an available encryption engine from the **Non HA Encryption Engines** table and a destination HA cluster from the **High Availability Clusters** table**.** Select **New HA Cluster** if you are creating a new cluster.

   NOTE
   If you are creating a new HA cluster, a dialog box displays requesting a name for the new HA cluster. HA Cluster names can have up to 31 characters. Letters, digits, and underscores are allowed.
4. Click the right arrow to add the encryption engine to the selected HA cluster.



To add an encryption engine (EE) to a high-availability cluster, select the EE on the left, the destination cluster on the right, and click the right arrow. To create a new cluster, select New HA Cluster as the destination. To remove an EE from a cluster, select the EE on the right and click the left arrow. To replace a cluster member with an EE from the left panel, select both EEs and click "<>". Replacing also moves configured targets from the original member to the replacement EE.

Non-HA Encryption Engines	Fabric	Link State
mace25 / Engine	10:0...	Up

High-Availability Clusters
- New HA Cluster
- new
  - mace241 / Engine (Online)

FIGURE 37 Encryption Group Properties dialog box - HA Clusters tab

To add the second encryption node to the HA cluster, perform the following procedure.

1. Select the desired HA cluster from the right panel.
2. Select the desired encryption engine to be added from the left panel.
3. Click the right arrow to add the encryption engine to the selected HA cluster.
4. Click **OK**.

Removing engines from an HA cluster

Removing the last engine from an HA cluster also removes the HA cluster.

If only one engine is removed from a two-engine cluster, you must either add another engine to the cluster, or remove the other engine.

1. Select **Configure > Encryption** from the menu task bar to display the **Encryption Center** dialog box (Refer to Figure 6 on page 14).
2. Select an encryption group from the **Encryption Center Devices** table, then select **Group > HA Cluster** from the menu task bar.

   The **Encryption Group Properties** dialog box displays, with the **HA Clusters** tab selected.
3. Select an engine from the **High Availability Clusters** table, then click the left arrow (Refer to Figure 37).
4. Either remove the second engine or add a replacement second engine, making sure all HA clusters have exactly two engines.
5. Click **OK**.

Swapping engines in an HA cluster

Swapping engines is useful when replacing hardware. Swapping engines is different from removing an engine and adding another because when you swap engines, the configured targets on the former HA cluster member are moved to the new HA cluster member.

1. Select **Configure** > **Encryption** from the menu task bar to display the **Encryption Center** dialog box.
2. Select an encryption group from the **Encryption Center Devices** table, then select **Group** > **HA Cluster** from the menu task bar

   The **Encryption Group Properties** dialog box displays, with the **HA Clusters** tab selected (Refer to Figure 37).

To swap engines, select one engine from the **High Availability Clusters** table and one unclustered engine from encryption engine from the **Non HA Encryption Engines** table, then click the double-arrow.

NOTE
The two engines being swapped must be in the same fabric.

Failback option

The **Failback** option determines the behavior when a failed encryption engine is restarted. When the first encryption engine comes back online, the encryption group's failback setting (auto or manual) determines how the encryption engine resumes encrypting and decrypting traffic to its encryption targets.

- In auto mode, when the first encryption engine restarts, it automatically resumes encrypting and decrypting traffic to its encryption targets.
- In manual mode, the second encryption engine continues handling the traffic until you manually invoke failback using the CLI or Management application, or until the second encryption engine fails. When the encryption engine recovers, it can automatically fail back its Crypto Target containers if the second encryption engine is not hosting them.

Invoking failback

To invoke failback to the restarted encryption engine from BNA, complete the following steps:

1. Select **Configure** > **Encryption** from the menu task bar to display the **Encryption Center** dialog box (Refer to Figure 6 on page 14).
2. Select an encryption group from the **Encryption Center Devices** table to which the encryption engine belongs, then click **Group** > **HA Clusters**.

   The **Encryption Group Properties** dialog box displays, with the **HA Clusters** tab selected (Refer to Figure 37).
3. Select the online encryption engine, then click **Failback**.
4. Click **OK**, then close the **Encryption Center** dialog box.

Configuring encryption storage targets

Adding an encryption target maps storage devices and hosts to virtual targets and virtual initiators within the encryption switch. The storage encryption wizard enables you to configure encryption for a storage device (target).

NOTE
It is recommended that you configure the host and target in the same zone before configuring them for encryption. If the host and target are not already in the same zone, you can still configure them for encryption, but you will need to configure them in the same zone before you can commit the changes. If you attempt to close the Encryption Targets dialog box without committing the changes, you are reminded of uncommitted changes in BNA.

The wizard steps are as follows:

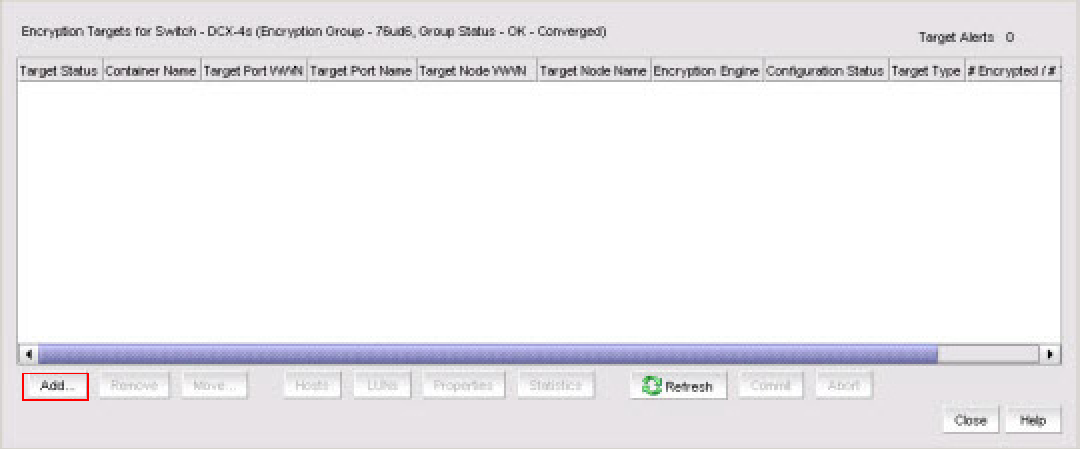
1. Select Encryption Engine
2. Select Target
3. Select Hosts
4. Name Container
5. Confirmation
6. Configuration Status
7. Important Instructions

Adding an encryption target

1. Select **Configure** > **Encryption** from the menu task bar to display the **Encryption Center** dialog box (Refer to Figure 6 on page 14).
2. Select a group, switch, or engine from the **Encryption Center Devices** table to which to add the target, then select **Group/Switch/Engine** > **Targets** from the menu task bar.

   NOTE
   You can also select a group, switch, or engine from the **Encryption Center Devices** table, then click the **Targets** icon.

   The **Encryption Targets** dialog box displays (Figure 38).



Encryption Targets for Switch - DCX-4s (Encryption Group - 78ud6, Group Status - OK - Converged)

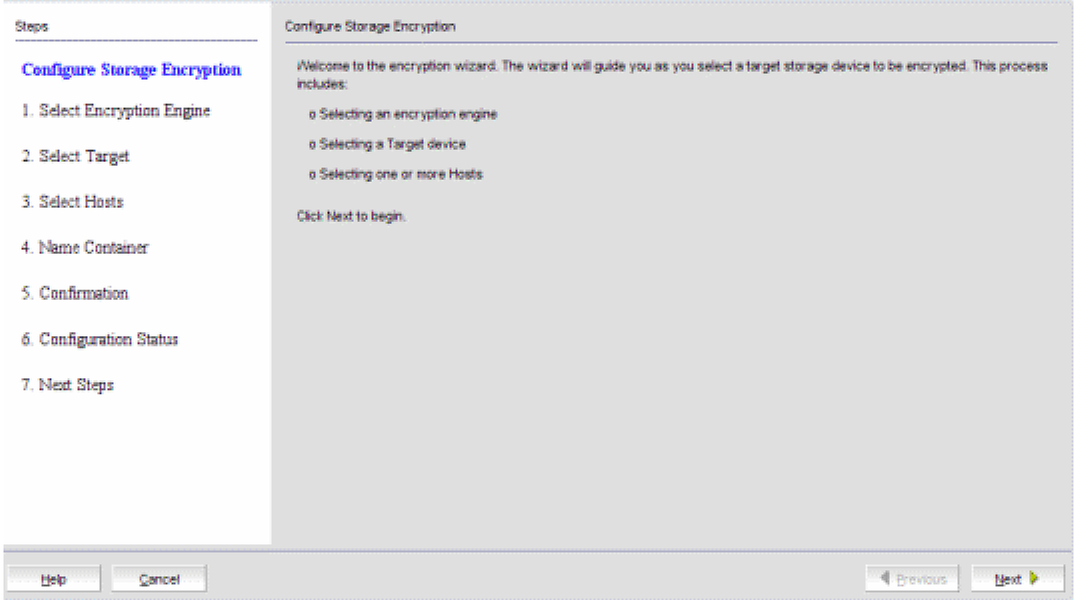
Target Alerts 0

Target Status	Container Name	Target Port WWN	Target Port Name	Target Node WWN	Target Node Name	Encryption Engine	Configuration Status	Target Type	# Encrypted / #

FIGURE 38 Encryption Targets dialog box

3. Click **Add**.

   The **Configure Storage Encryption** welcome screen displays (Figure 39).



Steps

Configure Storage Encryption

1. Select Encryption Engine
2. Select Target
3. Select Hosts
4. Name Container
5. Confirmation
6. Configuration Status
7. Next Steps

Configure Storage Encryption

Welcome to the encryption wizard. The wizard will guide you as you select a target storage device to be encrypted. This process includes:

o Selecting an encryption engine

o Selecting a Target device

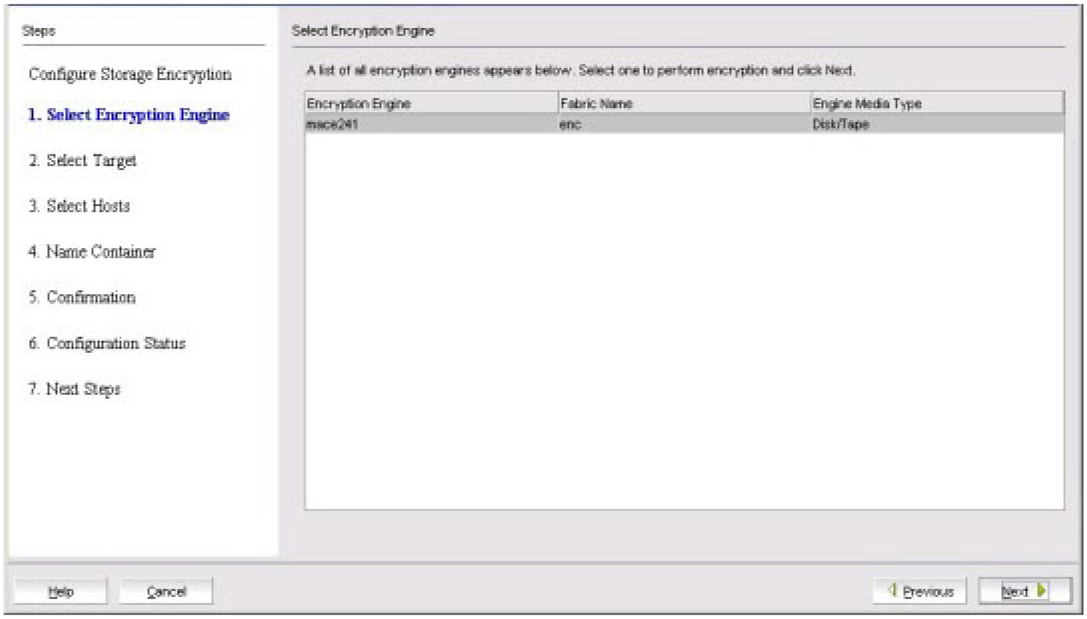
o Selecting one or more Hosts

Click Next to begin.

FIGURE 39 Configure Storage Encryption welcome screen

4. Click **Next**.

   The **Select Encryption Engine** dialog box displays (Figure 40).



Select Encryption Engine

A list of all encryption engines appears below. Select one to perform encryption and click Next.

Encryption Engine	Fabric Name	Engine Media Type
mace241	enc	Disk/Tape

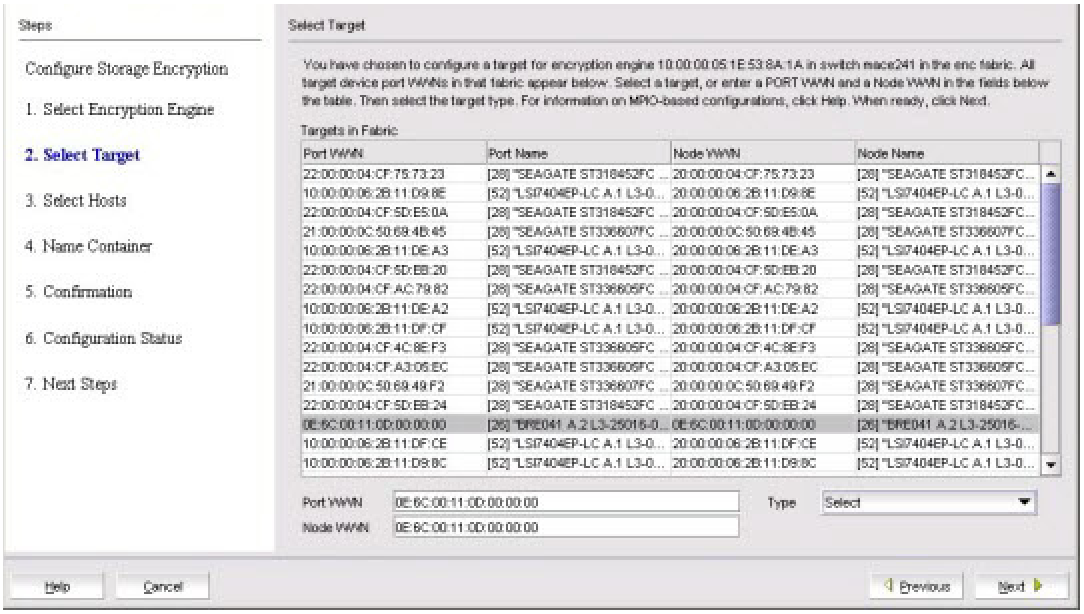
FIGURE 40 Select Encryption Engine dialog box

The dialog box contains the following information:

- **Encryption engine**: The name of the encryption engine. The list of engines depends on the scope being viewed:
  - If an encryption group was selected, the list includes all engines in the group.
  - If a switch was selected, the list includes all encryption engines for the switch.
  - If a single encryption engine was selected, the list contains only that engine.
- **Fabric Name**: The name of the fabric to which the selected encryption engine (blade or switch) is configured.
- **Engine Media Type:** The media type of the encryption engine. Options are: **Tape** and **Disk**.

5. Select the encryption engine (blade or switch) to configure, then click **Next.**

   The **Select Target** dialog box displays (Figure 41). The dialog box lists all target ports and target nodes in the same fabric as the encryption engine. The **Targets in Fabric** table does *not* show targets that are already configured in an encryption group.



Select Target

You have chosen to configure a target for encryption engine 10:00:00:05:1E:53:8A:1A in switch mace241 in the enc fabric. All target device port WWNs in that fabric appear below. Select a target, or enter a PORT WWN and a Node WWN in the fields below the table. Then select the target type. For information on MPIO-based configurations, click Help. When ready, click Next.

Targets in Fabric

Port WWN	Port Name	Node WWN	Node Name
22:00:00:04:CF:75:73:23	[28] "SEAGATE ST318452FC ...	20:00:00:04:CF:75:73:23	[28] "SEAGATE ST318452FC...
10:00:00:06:2B:11:D9:8E	[52] "LSI7404EP-LC A.1 L3-0...	20:00:00:06:2B:11:D9:8E	[52] "LSI7404EP-LC A.1 L3-0...
22:00:00:04:CF:5D:E5:0A	[28] "SEAGATE ST318452FC ...	20:00:00:04:CF:5D:E5:0A	[28] "SEAGATE ST318452FC...
21:00:00:0C:50:69:4B:45	[28] "SEAGATE ST336607FC ...	20:00:00:0C:50:69:4B:45	[28] "SEAGATE ST336607FC...
10:00:00:06:2B:11:DE:A3	[52] "LSI7404EP-LC A.1 L3-0...	20:00:00:06:2B:11:DE:A3	[52] "LSI7404EP-LC A.1 L3-0...
22:00:00:04:CF:5D:EB:20	[28] "SEAGATE ST318452FC ...	20:00:00:04:CF:5D:EB:20	[28] "SEAGATE ST318452FC...
22:00:00:04:CF:AC:79:82	[28] "SEAGATE ST336605FC ...	20:00:00:04:CF:AC:79:82	[28] "SEAGATE ST336605FC...
10:00:00:06:2B:11:DE:A2	[52] "LSI7404EP-LC A.1 L3-0...	20:00:00:06:2B:11:DE:A2	[52] "LSI7404EP-LC A.1 L3-0...
10:00:00:06:2B:11:DF:CF	[52] "LSI7404EP-LC A.1 L3-0...	20:00:00:06:2B:11:DF:CF	[52] "LSI7404EP-LC A.1 L3-0...
22:00:00:04:CF:4C:8E:F3	[28] "SEAGATE ST336605FC ...	20:00:00:04:CF:4C:8E:F3	[28] "SEAGATE ST336605FC...
22:00:00:04:CF:A3:05:EC	[28] "SEAGATE ST336605FC ...	20:00:00:04:CF:A3:05:EC	[28] "SEAGATE ST336605FC...
21:00:00:0C:50:69:49:F2	[28] "SEAGATE ST336607FC ...	20:00:00:0C:50:69:49:F2	[28] "SEAGATE ST336607FC...
22:00:00:04:CF:5D:EB:24	[28] "SEAGATE ST318452FC ...	20:00:00:04:CF:5D:EB:24	[28] "SEAGATE ST318452FC...
0E:6C:00:11:0D:00:00:00	[26] "BRE041 A.2 L3-25016-0...	0E:6C:00:11:0D:00:00:00	[26] "BRE041 A.2 L3-25016-...
10:00:00:06:2B:11:DF:CE	[52] "LSI7404EP-LC A.1 L3-0...	20:00:00:06:2B:11:DF:CE	[52] "LSI7404EP-LC A.1 L3-0...
10:00:00:06:2B:11:D9:8C	[52] "LSI7404EP-LC A.1 L3-0...	20:00:00:06:2B:11:D9:8C	[52] "LSI7404EP-LC A.1 L3-0...

Port WWN: 0E:6C:00:11:0D:00:00:00

Node WWN: 0E:6C:00:11:0D:00:00:00

Type: Select

FIGURE 41 **Select Target dialog box**

The dialog box contains the following information:

- **Target Port WWN:** The world wide name of the target port in the same fabric as the encryption engine.
- **Target Port Name:** The name of the target port in the same fabric as the encryption engine.
- **Target Node WWN**: The world wide name of the target node in the same fabric as the encryption engine.
- **Target Node Name:** The name of the target device.
- **Targets** list: Options are: **Tape** and **Disk**.

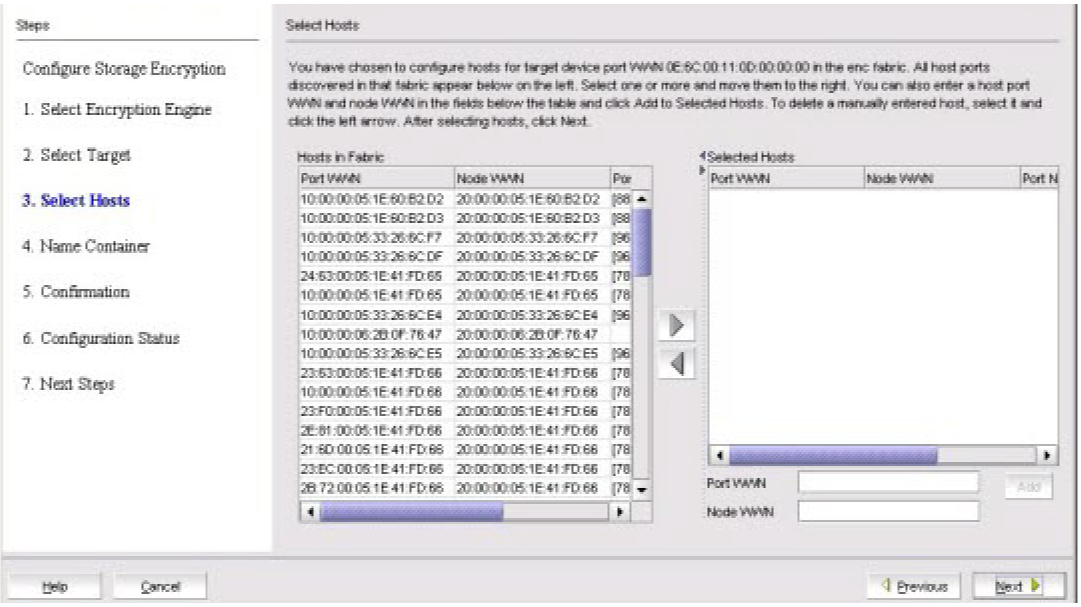
NOTE
The **Targets** list does not show targets that are already configured in the encryption group.

6. Select a target from the list. (The **Target Port WWN** and **Target Node WWN** fields contain all target information that displays when using the **nsShow** command.) You can also enter WWNs manually, for example, to specify a target that is not on the list.

7. Select a target type from the **Type** list, then click **Next.**

   The **Select Hosts** dialog box displays (Figure 42). You can configure hosts for selected target device ports. All hosts that are in the same fabric as the encryption engine are listed.

NOTE
The selected target and initiator port must be in the same zone, or an error will result.



You have chosen to configure hosts for target device port WWN 0E:6C:00:11:0D:00:00:00 in the enc fabric. All host ports discovered in that fabric appear below on the left. Select one or more and move them to the right. You can also enter a host port WWN and node WWN in the fields below the table and click Add to Selected Hosts. To delete a manually entered host, select it and click the left arrow. After selecting hosts, click Next.

Hosts in Fabric

Port WWN	Node WWN	Por
10:00:00:05:1E:60:B2:D2	20:00:00:05:1E:60:B2:D2	[88
10:00:00:05:1E:60:B2:D3	20:00:00:05:1E:60:B2:D3	[88
10:00:00:05:33:26:6C:F7	20:00:00:05:33:26:6C:F7	[96
10:00:00:05:33:26:6C:DF	20:00:00:05:33:26:6C:DF	[96
24:63:00:05:1E:41:FD:65	20:00:00:05:1E:41:FD:65	[78
10:00:00:05:1E:41:FD:65	20:00:00:05:1E:41:FD:65	[78
10:00:00:05:33:26:6C:E4	20:00:00:05:33:26:6C:E4	[96
10:00:00:06:2B:0F:76:47	20:00:00:06:2B:0F:76:47	
10:00:00:05:33:26:6C:E5	20:00:00:05:33:26:6C:E5	[96
23:63:00:05:1E:41:FD:66	20:00:00:05:1E:41:FD:66	[78
10:00:00:05:1E:41:FD:66	20:00:00:05:1E:41:FD:66	[78
23:F0:00:05:1E:41:FD:66	20:00:00:05:1E:41:FD:66	[78
2E:81:00:05:1E:41:FD:66	20:00:00:05:1E:41:FD:66	[78
21:6D:00:05:1E:41:FD:66	20:00:00:05:1E:41:FD:66	[78
23:EC:00:05:1E:41:FD:66	20:00:00:05:1E:41:FD:66	[78
2B:72:00:05:1E:41:FD:66	20:00:00:05:1E:41:FD:66	[78

Port WWN

Node WWN

FIGURE 42 Select Hosts dialog box

The dialog box contains the following information:

- Hosts in Fabric table: Lists the available hosts in the fabric.
- Selected Hosts table: Lists the hosts that have been selected to access the target.
- Port WWN: The world wide name of the host ports that are in the same fabric as the encryption engine.
- Node WWN: The world wide name of the host nodes that are in the same fabric as the encryption engine.
- Port Name: The user-assigned port name, if one exists; otherwise, the symbolic port name from the device.
- Port ID: The 24-bit Port ID of the host port.
- VI Port WWN: The world wide name of the virtual initiator port.
- VI Node WWN: The world wide name of the virtual initiator node.
- Host Name: The name of the hosts that are in the same fabric as the encryption engine.
- Port WWN text box: Type a world wide name for a host port.

  NOTE
  You must enter the host node world wide name before clicking Add, to add the WWN to the Selected Hosts table.

- Node WWN text box: Type a world wide name for a host node.

  NOTE
  You must also enter the host port world wide name before clicking Add to add the node WWN to the Selected Hosts table.

- Device Type: The device type indicated by the fabric's name service. The value is either Initiator or Initiator + Target.

- **Right arrow** button: Moves a host from the **Host in Fabric** table to the **Selected Hosts** table.
- **Left arrow** button: Removes a host from the **Selected Hosts** table.
- **Add** button: Click to manually add host port world wide names or host node world wide names to the **Selected Hosts** table.

8. Select hosts using either of the following methods:

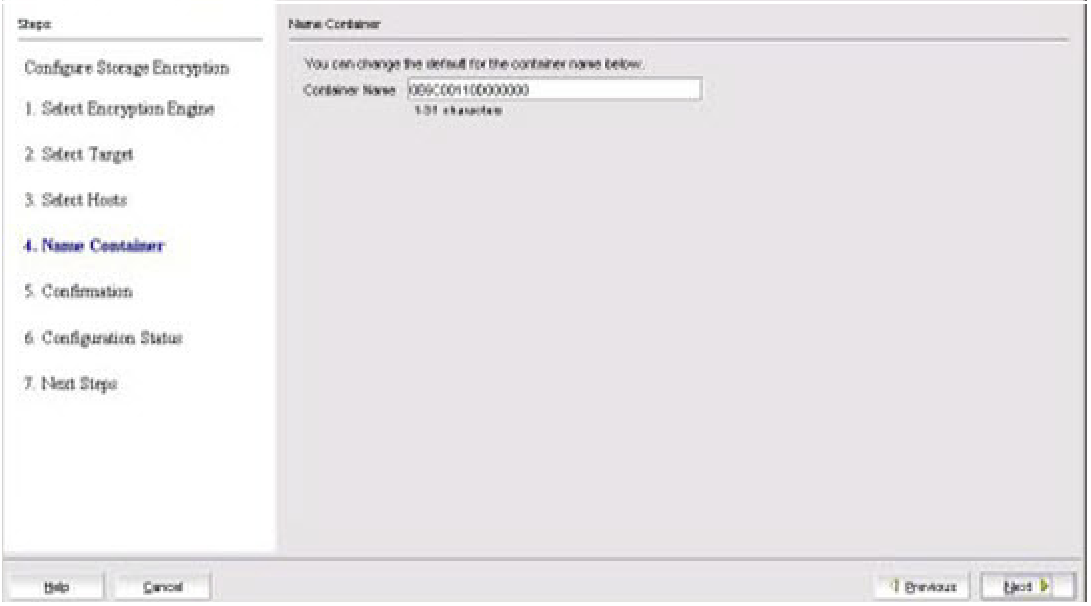
   a. Select a maximum of 1024 hosts from the **Hosts in Fabric** table, then click the right arrow to move the hosts to the **Selected Hosts** table. (The **Port WWN** column contains all target information that displays when using the **`nsshow`** command.)

   b. Manually enter world wide names in the **Port WWN** and **Node WWN** text boxes if the hosts are not included in the table. You must fill in both the Port WWN and the Node WWN. Click **Add** to move the host to the **Selected Hosts** table.

9. Click **Next**.

   The **Name Container** dialog box displays (Figure 43). You can specify a name for the target container that is created in the encryption engine to hold the target configuration data. The name is only needed when configuring the storage using the command line interface (CLI).

   The container name defaults to the target WWPN. You can, however, rename the container name. Target container names can have up to 31 characters. Letters, digits, and underscores are allowed.



You can change the default for the container name below.

Container Name 009C001100000000

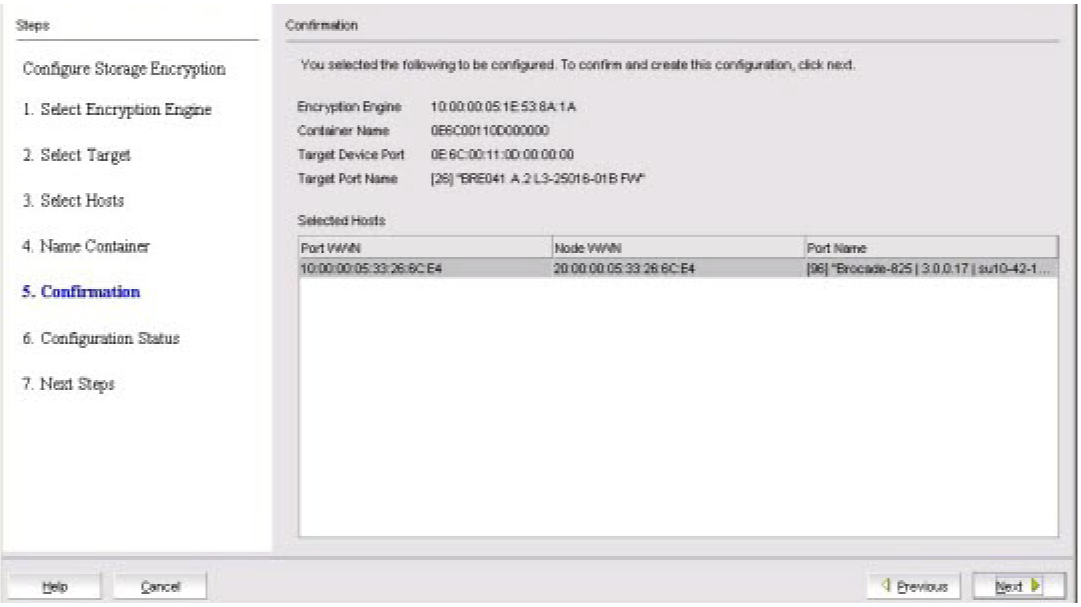
1-31 characters

FIGURE 43 Name Container dialog box

10. Enter the container name. The container name is a logical encryption name to specify a name other than the default. You can use a maximum of 31 characters. Letters, digits, and underscores are allowed.

11. Click **Next**.

    The **Confirmation** screen displays (Figure 44). The confirmation screen confirms and completes configuration of encryption engines, targets, and hosts.



Confirmation

You selected the following to be configured. To confirm and create this configuration, click next.

Encryption Engine	10:00:00:05:1E:53:8A:1A
Container Name	0E6C00110D000000
Target Device Port	0E:6C:00:11:0D:00:00:00
Target Port Name	[26] "BRE041 A 2 L3-25016-01B FW"

Selected Hosts

Port WWN	Node WWN	Port Name
10:00:00:05:33:26:6C:E4	20:00:00:05:33:26:6C:E4	[96] "Brocade-825 | 3.0.0.17 | su10-42-1...

FIGURE 44 Confirmation dialog box

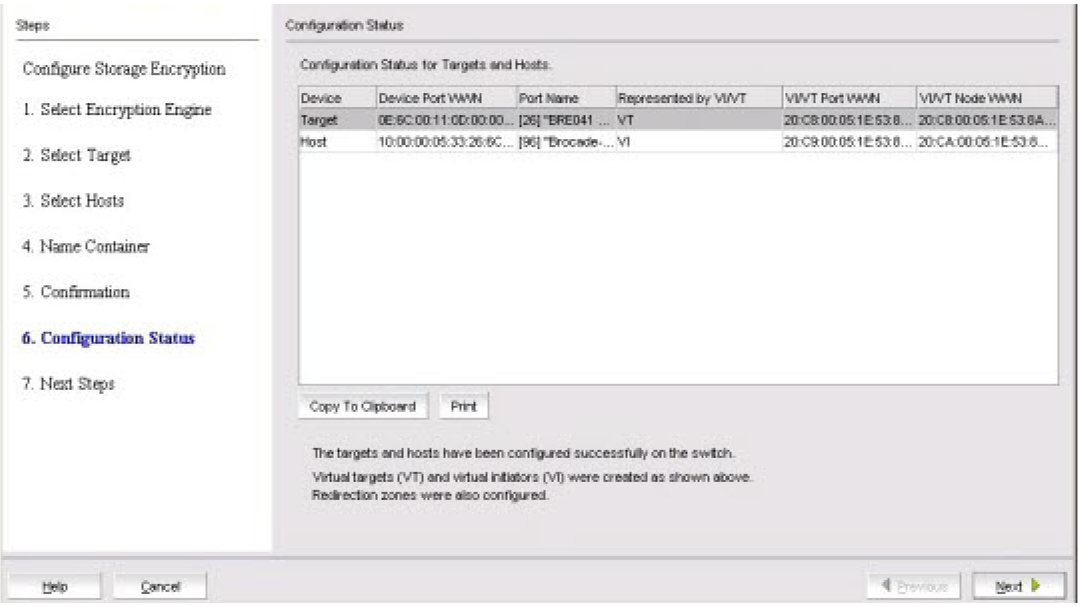
The screen contains the following information:

- Encryption Engine: The slot location of the encryption engine.
- Container Name: The logical encryption name used to map storage targets and hosts to virtual targets and virtual initiators.
- Target Device Port: The world wide name of the target device port.
- Host Node WWN: The world wide name of the host node.
- Host Port WWN: The world wide name of the host port.
- Host Name: The name of the host.

12. Verify the information is correct, then click **Next**, which creates the configuration.

    The **Configuration Status** screen displays (Figure 45), which shows the status of the new container configuration. The target and host that are configured in the target container are listed, as well as the virtual targets (VT) and virtual initiators (VI).

NOTE
If you can view the VI/VT Port WWNs and VI/VT Node WWNs, the container has been successfully added to the switch.



Configuration Status for Targets and Hosts.

Device	Device Port WWN	Port Name	Represented by VI/VT	VI/VT Port WWN	VI/VT Node WWN
Target	0E:6C:00:11:0D:00:00...	[26] "BRE041 ...	VT	20:C8:00:05:1E:53:8...	20:C8:00:05:1E:53:8A...
Host	10:00:00:05:33:26:6C...	[96] "Brocade-...	VI	20:C9:00:05:1E:53:8...	20:CA:00:05:1E:53:8...

The targets and hosts have been configured successfully on the switch.

Virtual targets (VT) and virtual initiators (VI) were created as shown above.
Redirection zones were also configured.

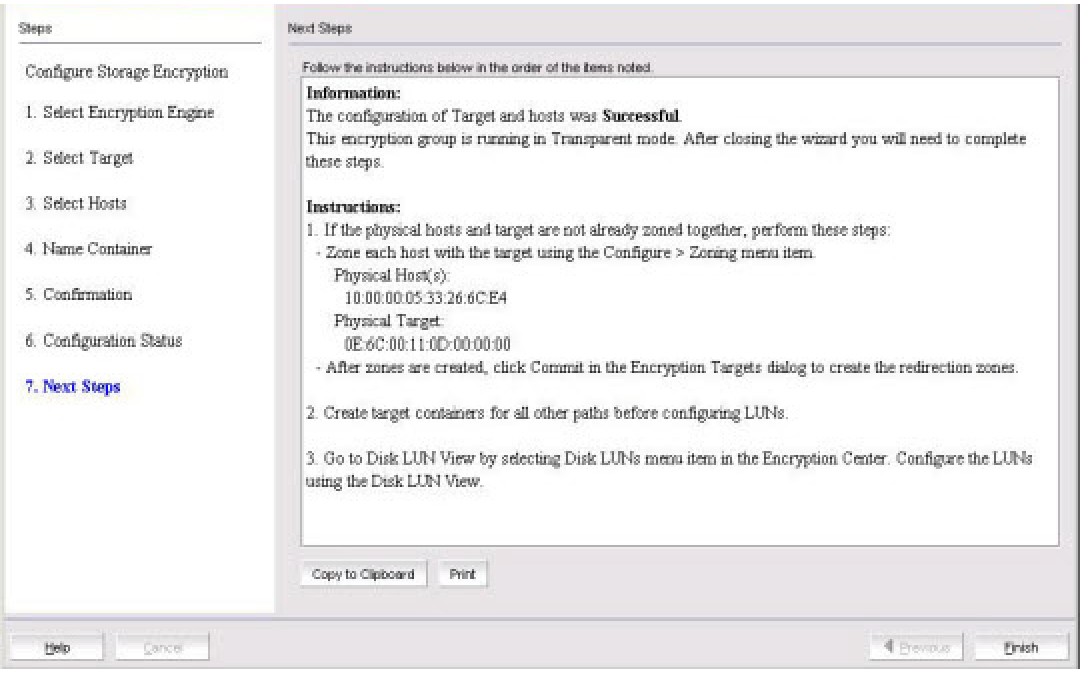
FIGURE 45 **Configuration Status screen**

The screen contains the following information:

- Device: The device type (target or host).
- Device Port WWN: The port world wide name.
- Represented by VI/VT: The virtual target (VT) mapped to the physical target or virtual initiator (VI) representing the host.
- VI/VT Port WWN: The port world wide name of the virtual target or virtual initiator.
- VI/VT Node WWN: The node world wide name of the virtual target or virtual initiator.

13. Review any post-configuration instructions or messages, which you can copy to a clipboard or print for later, then click Next.

    The Next Steps screen displays (Figure 46). Post-configuration instructions for installing public key certificates for the encryption switch are displayed. These instructions are specific to the key vault type.



Configure Storage Encryption

1. Select Encryption Engine
2. Select Target
3. Select Hosts
4. Name Container
5. Confirmation
6. Configuration Status
7. Next Steps

Follow the instructions below in the order of the items noted.

Information:
The configuration of Target and hosts was **Successful**.
This encryption group is running in Transparent mode. After closing the wizard you will need to complete these steps.

Instructions:
1. If the physical hosts and target are not already zoned together, perform these steps:
 - Zone each host with the target using the Configure > Zoning menu item.
 Physical Host(s):
 10:00:00:05:33:26:6C:E4
 Physical Target:
 0E:6C:00:11:0D:00:00:00
 - After zones are created, click Commit in the Encryption Targets dialog to create the redirection zones.

2. Create target containers for all other paths before configuring LUNs.

3. Go to Disk LUN View by selecting Disk LUNs menu item in the Encryption Center. Configure the LUNs using the Disk LUN View.

FIGURE 46 Next Steps screen

The screen contains the following information:

- **Important Instructions:** Instructions about post-configuration tasks you must complete after you close the wizard. For example, you must zone the physical hosts and the target together and then you encrypt the LUNs using the **Storage Device LUNs** dialog box.
- **Copy to Clipboard** button: Saves a copy of the instructions.
- **Print** button: Prints the configuration.

14. Review the post-configuration instructions, which you can copy to a clipboard or print for later, then click **Finish** to exit the **Configure Switch Encryption** wizard.

Configuring hosts for encryption targets

Use the **Encryption Target Hosts** dialog box to edit (add or remove) hosts for an encrypted target.

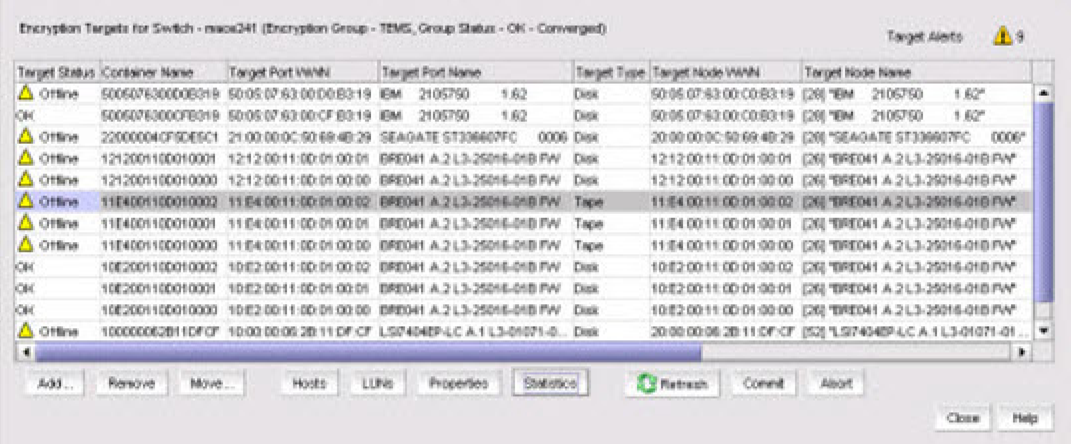
NOTE
Hosts are normally selected as part of the **Configure Switch Encryption** wizard, but you can also edit hosts later using the **Encryption Target Hosts** dialog box.

1. Select **Configure** > **Encryption** from the menu task bar to display the **Encryption Center** dialog box (Refer to Figure 6 on page 14).

2. Select a group, switch, or engine from the **Encryption Center Devices** table that contains the storage device to be configured, then select **Group/Switch/Engine** > **Targets** from the menu task bar.

NOTE
You can also select a group, switch, or engine from the **Encryption Center Devices** table, then click the **Targets** icon.

The **Encryption Targets** dialog box displays (Figure 47).



Encryption Targets for Switch - maca241 (Encryption Group - TEMS_Group Status - OK - Converged)

Target Alerts ⚠ 9

Target Status	Container Name	Target Port WWN	Target Port Name	Target Type	Target Node WWN	Target Node Name
⚠ Offline	5005076300D0B319	50:05:07:63:00:D0:B3:19	IBM 2105750 1.62	Disk	50:05:07:63:00:C0:B3:19	[28] "IBM 2105750 1.62"
OK	5005076300CFB319	50:05:07:63:00:CF:B3:19	IBM 2105750 1.62	Disk	50:05:07:63:00:C0:B3:19	[28] "IBM 2105750 1.62"
⚠ Offline	22000004CF9DE5C1	21:00:00:0C:50:69:4B:29	SEAGATE ST336607FC 0006	Disk	20:00:00:0C:50:69:4B:29	[28] "SEAGATE ST336607FC 0006"
⚠ Offline	1212001100010001	12:12:00:11:00:01:00:01	BRE041 A.2 L3-25016-01B FW	Disk	12:12:00:11:00:01:00:01	[26] "BRE041 A.2 L3-25016-01B FW"
⚠ Offline	1212001100010000	12:12:00:11:00:01:00:00	BRE041 A.2 L3-25016-01B FW	Disk	12:12:00:11:00:01:00:00	[26] "BRE041 A.2 L3-25016-01B FW"
⚠ Offline	11E4001100010002	11:E4:00:11:00:01:00:02	BRE041 A.2 L3-25016-01B FW	Tape	11:E4:00:11:00:01:00:02	[26] "BRE041 A.2 L3-25016-01B FW"
⚠ Offline	11E4001100010001	11:E4:00:11:00:01:00:01	BRE041 A.2 L3-25016-01B FW	Tape	11:E4:00:11:00:01:00:01	[26] "BRE041 A.2 L3-25016-01B FW"
⚠ Offline	11E4001100010000	11:E4:00:11:00:01:00:00	BRE041 A.2 L3-25016-01B FW	Tape	11:E4:00:11:00:01:00:00	[26] "BRE041 A.2 L3-25016-01B FW"
OK	10E2001100010002	10:E2:00:11:00:01:00:02	BRE041 A.2 L3-25016-01B FW	Disk	10:E2:00:11:00:01:00:02	[26] "BRE041 A.2 L3-25016-01B FW"
OK	10E2001100010001	10:E2:00:11:00:01:00:01	BRE041 A.2 L3-25016-01B FW	Disk	10:E2:00:11:00:01:00:01	[26] "BRE041 A.2 L3-25016-01B FW"
OK	10E2001100010000	10:E2:00:11:00:01:00:00	BRE041 A.2 L3-25016-01B FW	Disk	10:E2:00:11:00:01:00:00	[26] "BRE041 A.2 L3-25016-01B FW"
⚠ Offline	100000062B11DFCF	10:00:00:06:2B:11:DF:CF	LSI7404EP-LC A.1 L3-01071-0...	Disk	20:00:00:06:2B:11:DF:CF	[52] "LSI7404EP-LC A.1 L3-01071-01 ...

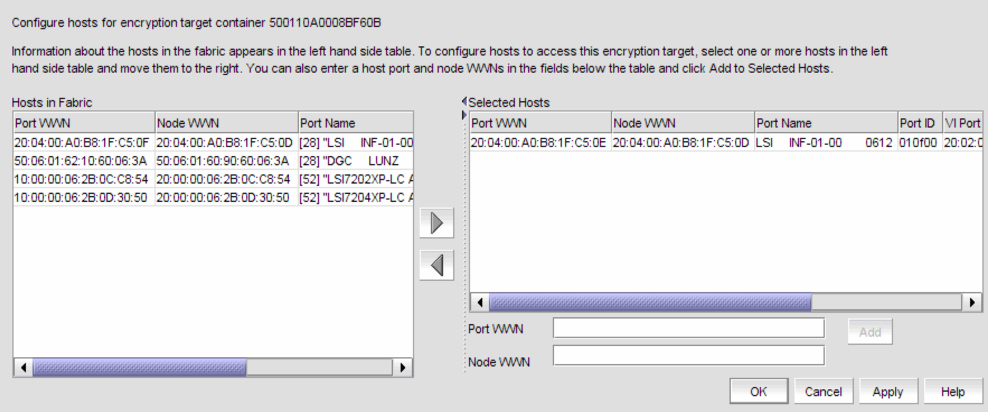
FIGURE 47 Encryption Targets dialog box

3. Select a target storage device from the list, then click **Hosts**.

   The **Encryption Target Hosts** dialog box displays (Figure 48). The **Hosts in Fabric** table lists the configured hosts in a fabric.

   The table displays the following information:

   - **Port WWN**: The world wide name of the host ports that are in the same fabric as the encryption engine.
   - **Node WWN**: The world wide name of the host nodes that are in the same fabric as the encryption engine.
   - **Port Name**: The name of the hosts that are in the same fabric as the encryption engine.
   - **Port ID**: Displays the 24-bit port ID (PID) of the host port in both the **Host Ports in Fabric** table and the **Selected Hosts** table.



Configure hosts for encryption target container 500110A0008BF60B

Information about the hosts in the fabric appears in the left hand side table. To configure hosts to access this encryption target, select one or more hosts in the left hand side table and move them to the right. You can also enter a host port and node WWNs in the fields below the table and click Add to Selected Hosts.

Hosts in Fabric

Port WWN	Node WWN	Port Name
20:04:00:A0:B8:1F:C5:0F	20:04:00:A0:B8:1F:C5:0D	[28] "LSI INF-01-00
50:06:01:62:10:60:06:3A	50:06:01:60:90:60:06:3A	[28] "DGC LUNZ
10:00:00:06:2B:0C:C8:54	20:00:00:06:2B:0C:C8:54	[52] "LSI7202XP-LC A
10:00:00:06:2B:0D:30:50	20:00:00:06:2B:0D:30:50	[52] "LSI7204XP-LC A

Selected Hosts

Port WWN	Node WWN	Port Name	Port ID	VI Port
20:04:00:A0:B8:1F:C5:0E	20:04:00:A0:B8:1F:C5:0D	LSI INF-01-00 0612	010f00	20:02:0

Port WWN

Node WWN

FIGURE 48 Encryption Target Hosts dialog box

NOTE
Both the **Host Ports in Fabric** table and the **Selected Hosts** table now contain a **Port ID** column to display the 24-bit PID of the host port.

4. Select one or more hosts in a fabric using either of the following methods:

   a. Select a maximum of 1024 hosts from the **Hosts in Fabric** table, then click the right arrow to move the hosts to the **Selected Hosts** table. (The **Port WWN** column contains all target information that displays when using the **`nsshow`** command.)

   b. Manually enter world wide names in the **Port WWN** and **Node WWN** text boxes if the hosts are not included in the table. You must fill in both the Port WWN and the Node WWN. Click the **Right-arrow** button to move the host to the **Selected Hosts** table.

NOTE
The selected host and target must be in the same zone, or an error will result.

The Selected Hosts table lists the following:

- **Port WWN**: The selected host port's world wide name.
- **Node WWN**: The selected host node's world wide name.
- **Port Name**: The name of the host selected to access the encryption target.
- **Port WWN** text box: Type a world wide name for a host port, and click the Add to Selected Hosts button to add to the Selected Hosts table.
- **Port ID**: Displays the 24-bit port ID (PID) of the host port in both the Host Ports in Fabric table and the Selected Hosts table.
- **VI Port WWN**: The world wide name of the virtual initiator port.
- **VI Node WWN**: The world wide name of the virtual initiator node.

NOTE
To remove an encryption engine from the **Selected Hosts** table, select the engine(s), then click the **Left-arrow** button.

5. Click **OK** or **Apply** to apply your changes.

Adding target disk LUNs for encryption

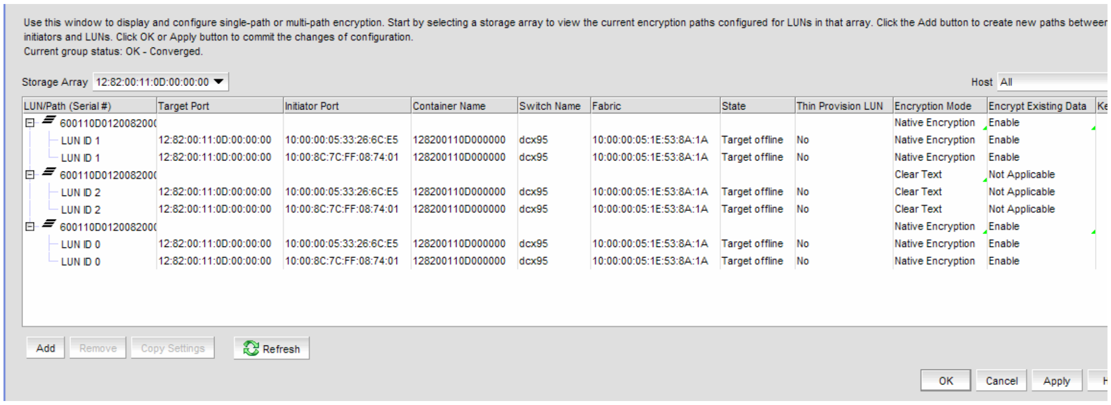
You can add a new path to an existing disk LUN or add a new LUN and path by launching the **Add New Path** wizard.

NOTE
Before you can add a target disk LUN for encryption, you must first configure the Storage Arrays. For more information, see "Configuring storage arrays" on page 71.

Complete the following steps to add a target disk LUN:

1. Select **Configure > Encryption** from the menu task bar to display the **Encryption Center** dialog box.
2. Select a group, switch, or engine from the **Encryption Center Devices** table, then select **Group/Switch/Engine > Disk LUNs** from the menu task bar.

   The **Encryption Disk LUN View** dialog box displays (Figure 49).



Use this window to display and configure single-path or multi-path encryption. Start by selecting a storage array to view the current encryption paths configured for LUNs in that array. Click the Add button to create new paths between initiators and LUNs. Click OK or Apply button to commit the changes of configuration.
Current group status: OK - Converged.

Storage Array 12:82:00:11:0D:00:00:00

Host All

LUN/Path (Serial #)	Target Port	Initiator Port	Container Name	Switch Name	Fabric	State	Thin Provision LUN	Encryption Mode	Encrypt Existing Data	Ke
600110D012008200(								Native Encryption	Enable	
LUN ID 1	12:82:00:11:0D:00:00:00	10:00:00:05:33:26:6C:E5	128200110D000000	dcx95	10:00:00:05:1E:53:8A:1A	Target offline	No	Native Encryption	Enable	
LUN ID 1	12:82:00:11:0D:00:00:00	10:00:8C:7C:FF:08:74:01	128200110D000000	dcx95	10:00:00:05:1E:53:8A:1A	Target offline	No	Native Encryption	Enable	
600110D012008200(								Clear Text	Not Applicable	
LUN ID 2	12:82:00:11:0D:00:00:00	10:00:00:05:33:26:6C:E5	128200110D000000	dcx95	10:00:00:05:1E:53:8A:1A	Target offline	No	Clear Text	Not Applicable	
LUN ID 2	12:82:00:11:0D:00:00:00	10:00:8C:7C:FF:08:74:01	128200110D000000	dcx95	10:00:00:05:1E:53:8A:1A	Target offline	No	Clear Text	Not Applicable	
600110D012008200(								Native Encryption	Enable	
LUN ID 0	12:82:00:11:0D:00:00:00	10:00:00:05:33:26:6C:E5	128200110D000000	dcx95	10:00:00:05:1E:53:8A:1A	Target offline	No	Native Encryption	Enable	
LUN ID 0	12:82:00:11:0D:00:00:00	10:00:8C:7C:FF:08:74:01	128200110D000000	dcx95	10:00:00:05:1E:53:8A:1A	Target offline	No	Native Encryption	Enable	

FIGURE 49 Encryption Disk LUN View dialog box

The dialog box provides a convenient way to view and manage disk LUNs that are provisioned from different hosts, identify conflicts between configuration policies on storage systems, and to provide a launching point for the **Add New Path** wizard for configuring multiple I/O paths to the LUN.

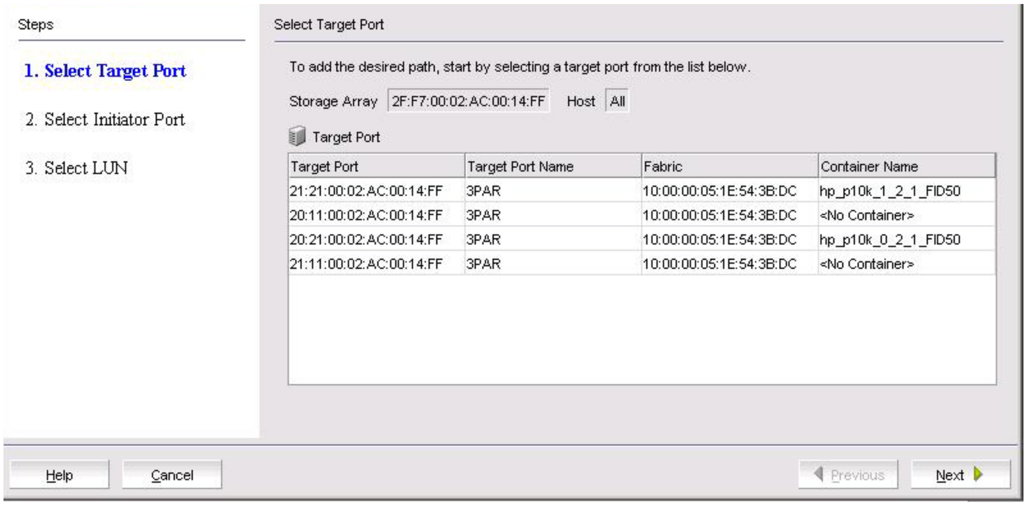
The dialog box contains the following information:

- **Storage Array** selector: Determines which LUN paths are displayed in the table. Enables you to select a storage array from the LUN view prior to launching the **Add New Path** wizard. Only ports that belong to at least one target container are listed.
- **Host** selector: Used to select a host from the LUN view prior to launching the **Add New Path** wizard. Only ports that belong to at least one target container are listed.
- **Encryption path** table: Should be LUN/Path identified by the following:
  - LUN Path Serial #
  - Target Port
  - Initiator Port
  - Container Name
  - Switch Name

    - Fabric
    - State
    - Thin Provision LUN
    - Encryption Mode
    - Encrypt Existing Data
    - Key ID
- **Remove** button: Removes a selected entry from the table.

3. Click **Add** to launch the **Add New Path** wizard.

   The **Select Target Port** dialog box displays (Figure 50).



To add the desired path, start by selecting a target port from the list below.

Storage Array 2F:F7:00:02:AC:00:14:FF Host All

Target Port	Target Port Name	Fabric	Container Name
21:21:00:02:AC:00:14:FF	3PAR	10:00:00:05:1E:54:3B:DC	hp_p10k_1_2_1_FID50
20:11:00:02:AC:00:14:FF	3PAR	10:00:00:05:1E:54:3B:DC	<No Container>
20:21:00:02:AC:00:14:FF	3PAR	10:00:00:05:1E:54:3B:DC	hp_p10k_0_2_1_FID50
21:11:00:02:AC:00:14:FF	3PAR	10:00:00:05:1E:54:3B:DC	<No Container>

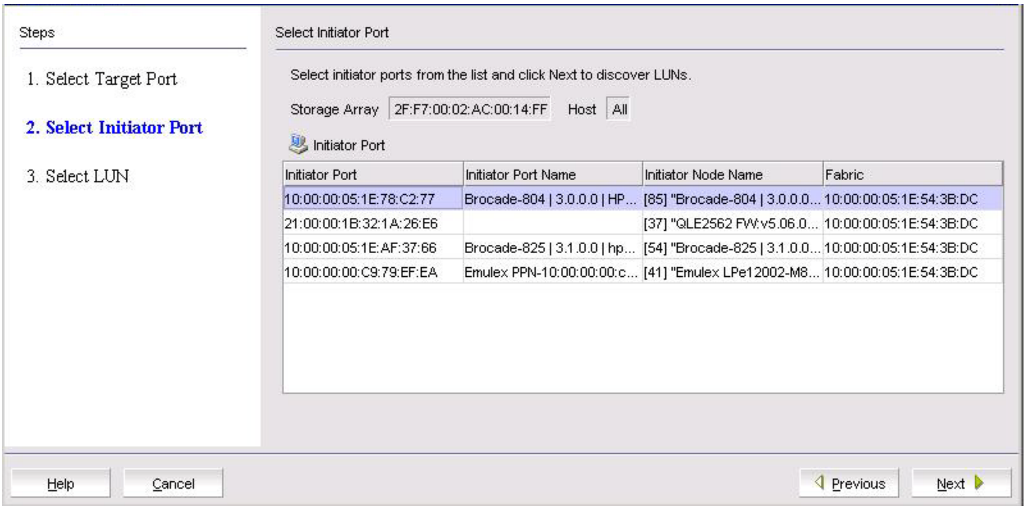
FIGURE 50 **Select Target Port dialog box**

The dialog box is used to select a target port when configuring multiple I/O paths to a disk LUN. The dialog box contains the following information:

- **Storage Array:** The Storage Array selected from the LUN view prior to launching the **Add New Path** wizard.
- **Host**: The host selected from the LUN view prior to launching the **Add New Path** wizard.
- **Target Port** table: Lists target ports using the following identifiers:
    - Target Port
    - Target Port Name
    - Fabric
    - Container Name

4. Select the target port from the **Target Port** table, then click **Next**.

   The **Select Initiator Port** dialog box displays (Figure 51).



Select Initiator Port

Select initiator ports from the list and click Next to discover LUNs.

Storage Array 2F:F7:00:02:AC:00:14:FF Host All

Initiator Port

Initiator Port	Initiator Port Name	Initiator Node Name	Fabric
10:00:00:05:1E:78:C2:77	Brocade-804 | 3.0.0.0 | HP...	[85] "Brocade-804 | 3.0.0.0...	10:00:00:05:1E:54:3B:DC
21:00:00:1B:32:1A:26:E6		[37] "QLE2562 FW:v5.06.0...	10:00:00:05:1E:54:3B:DC
10:00:00:05:1E:AF:37:66	Brocade-825 | 3.1.0.0 | hp...	[54] "Brocade-825 | 3.1.0.0...	10:00:00:05:1E:54:3B:DC
10:00:00:00:C9:79:EF:EA	Emulex PPN-10:00:00:00:c...	[41] "Emulex LPe12002-M8...	10:00:00:05:1E:54:3B:DC

FIGURE 51 **Select Initiator Port dialog box**

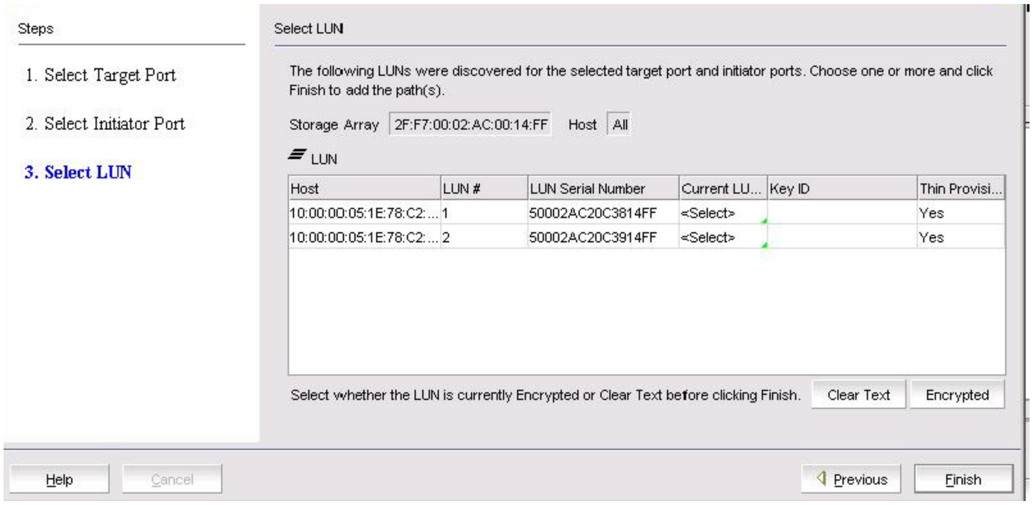
The dialog box is used to select an initiator port when configuring multiple I/O paths to a disk LUN. The dialog box contains the following information:

- **Storage Array**: Displays the storage array that was selected from the LUN view prior to launching the wizard.
- **Host**: The host selected from the LUN view prior to launching the wizard.
- **Initiator Port** table: Lists initiator ports using the following identifiers:
  - **Initiator Port**
  - **Initiator Port Name**
  - **Initiator Node Name**
  - **Fabric**

5. Select the initiator port from the **Initiator Port** table, then click **Next.**

   LUN discovery is launched and a progress bar displays. There are four possible outcomes:

   - A message displays indicating no LUNs were discovered. Click **OK** to dismiss the message and exit the wizard.
   - A message displays indicating LUNs have been discovered, but are already configured. Click **OK** to dismiss the message and exit the wizard.
   - A message displays indicating that the target is not in the right state for discovering LUNs. Click **OK** to dismiss the message and exit the wizard.
   - The **Select LUN** dialog box displays (Figure 52), which lists discovered LUNs that are available.



The following LUNs were discovered for the selected target port and initiator ports. Choose one or more and click Finish to add the path(s).

Storage Array 2F:F7:00:02:AC:00:14:FF Host All

Host	LUN #	LUN Serial Number	Current LU...	Key ID	Thin Provisi...
10:00:00:05:1E:78:C2:...	1	50002AC20C3814FF	<Select>		Yes
10:00:00:05:1E:78:C2:...	2	50002AC20C3914FF	<Select>		Yes

Select whether the LUN is currently Encrypted or Clear Text before clicking Finish.

FIGURE 52 Select LUN dialog box

The dialog box is used to select a LUN when configuring multiple I/O paths to a disk LUN. The dialog box contains the following information:

- Storage Array: The Storage Array selected from the LUN view prior to launching the Add New Path wizard.
- Host: The host elected from the LUN view prior to launching the Add New Path wizard.
- LUN table: Available LUNs identified by the following:
  - Host
  - LUN Number
  - LUN Serial Number
  - Current LUN State: Options are **Encrypted**, which is automatically selected if the LUN has a key ID; **Clear Text**, and **<select>** for LUNs without a key ID. User selection is required.
- Key ID: Identifies the key ID for discovered LUNs.
- Thin Provision LUN: Identifies if the new LUN is a thin provisioned LUN.Options are **Yes**, **No**, **Unknown**, or **Not Applicable**.

NOTE
Thin provision support is limited to Brocade-tested storage arrays. The thin provisioned LUN status will be displayed as **Yes** for supported storage arrays only.

- New LUN: Displayed only if remote replication is enabled.

6. Select the LUN from **LUN** list.
7. Set the **Current LUN State** as required. If the LUN already has an existing key ID, the **Current LUN State** field is automatically set to **Encrypted**. You can accept the automatically assigned state or change this value if desired.
8. If **REPL Support** was enabled by the **Configure Switch Encryption** wizard, a **New LUN** check box is presented and enabled by default. If this LUN is to be paired with another LUN for SRDF data replication, the **New LUN** option must be enabled. Refer to “Metadata requirements and remote replication” for information about how this option works. If **REPL support** was not enabled, this check box is not displayed.

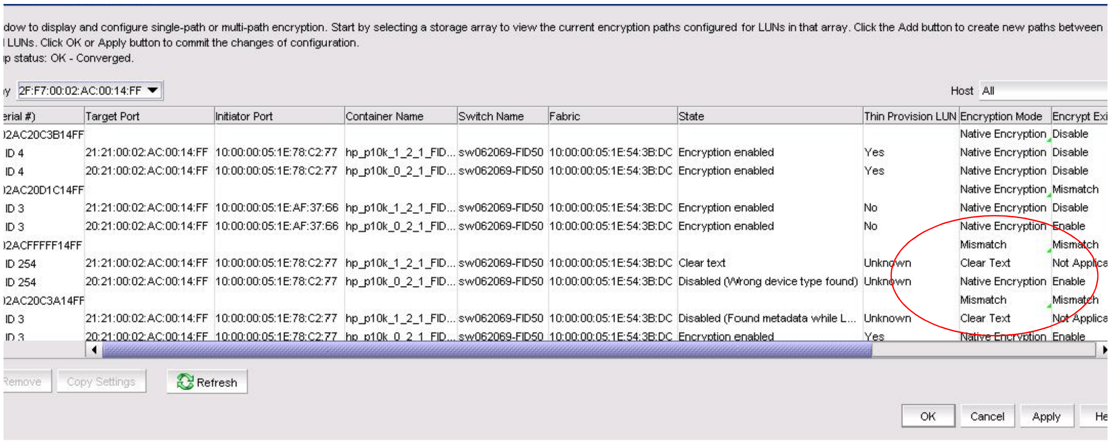
9. Click **Finish**.

   The new LUN path is added to the **Encryption Disk LUN View** table.

10. Click **OK** on the LUN view to commit the operation.

NOTE
With the introduction of Fabric OS v7.1.0, the maximum number of uncommitted configuration changes per disk LUN (or maximum paths to a LUN) is 512 transactions. The 512 LUN operations can be for the same LUN or be subjected to 25 distinct LUNs. This change of restriction in commit limit is applicable when using BNA only. Earlier Fabric OS versions allowed a maximum of 25 uncommitted changes per disk LUN. Adding or modifying more than 25 paths on the same LUN is not recommended unless the LUN is encrypted.

In environments where there are multiple paths to the same LUNs, it is critical that the same LUN policies are configured on all instances of the LUN. Be sure to return to the **Encryption Disk LUN View** dialog box to determine if there are configuration mismatches. Check under **Encryption Mode** for any entries showing **Mismatch.** To correct the mismatch, click the incorrect mode to display the options, then select the correct mode (Figure 53).



dow to display and configure single-path or multi-path encryption. Start by selecting a storage array to view the current encryption paths configured for LUNs in that array. Click the Add button to create new paths between
l LUNs. Click OK or Apply button to commit the changes of configuration.
p status: OK - Converged.

y 2F:F7:00:02:AC:00:14:FF ▼ Host All

erial #)	Target Port	Initiator Port	Container Name	Switch Name	Fabric	State	Thin Provision LUN	Encryption Mode	Encrypt Exi
)2AC20C3B14FF								Native Encryption	Disable
ID 4	21:21:00:02:AC:00:14:FF	10:00:00:05:1E:78:C2:77	hp_p10k_1_2_1_FID...	sw062069-FID50	10:00:00:05:1E:54:3B:DC	Encryption enabled	Yes	Native Encryption	Disable
ID 4	20:21:00:02:AC:00:14:FF	10:00:00:05:1E:78:C2:77	hp_p10k_0_2_1_FID...	sw062069-FID50	10:00:00:05:1E:54:3B:DC	Encryption enabled	Yes	Native Encryption	Disable
)2AC20D1C14FF								Native Encryption	Mismatch
ID 3	21:21:00:02:AC:00:14:FF	10:00:00:05:1E:AF:37:66	hp_p10k_1_2_1_FID...	sw062069-FID50	10:00:00:05:1E:54:3B:DC	Encryption enabled	No	Native Encryption	Disable
ID 3	20:21:00:02:AC:00:14:FF	10:00:00:05:1E:AF:37:66	hp_p10k_0_2_1_FID...	sw062069-FID50	10:00:00:05:1E:54:3B:DC	Encryption enabled	No	Native Encryption	Enable
)2ACFFFFF14FF								Mismatch	Mismatch
ID 254	21:21:00:02:AC:00:14:FF	10:00:00:05:1E:78:C2:77	hp_p10k_1_2_1_FID...	sw062069-FID50	10:00:00:05:1E:54:3B:DC	Clear text	Unknown	Clear Text	Not Applica
ID 254	20:21:00:02:AC:00:14:FF	10:00:00:05:1E:78:C2:77	hp_p10k_0_2_1_FID...	sw062069-FID50	10:00:00:05:1E:54:3B:DC	Disabled (Wrong device type found)	Unknown	Native Encryption	Enable
)2AC20C3A14FF								Mismatch	Mismatch
ID 3	21:21:00:02:AC:00:14:FF	10:00:00:05:1E:78:C2:77	hp_p10k_1_2_1_FID...	sw062069-FID50	10:00:00:05:1E:54:3B:DC	Disabled (Found metadata while L...	Unknown	Clear Text	Not Applica
ID 3	20:21:00:02:AC:00:14:FF	10:00:00:05:1E:78:C2:77	hp_p10k_0_2_1_FID...	sw062069-FID50	10:00:00:05:1E:54:3B:DC	Encryption enabled	Yes	Native Encryption	Enable

FIGURE 53 Correcting an Encryption Mode Mismatch

When you correct a policy on a LUN, it is automatically selected for all paths to the selected LUN. When you modify LUN policies, a **Modify** icon displays to identify the modified LUN entry.

11. Click **OK** or **Apply** to apply the changes.

Configuring storage arrays

The Storage Array contains a list of storage ports that will be used later in the LUN centric view. You must assign storage ports from the same storage array for multi-path I/O purposes. On the LUN centric view, storage ports in the same storage array are used to get the associated CryptoTarget containers and initiators from the database. Storage ports that are not assigned to any storage array but are within the fabrics of the encryption group will be listed as a single target port on the LUN centric view. Storage Arrays are configured using the **Storage Port Mapping** dialog box. You will need to:

1. Configure target and zone initiator ports in the same zone in order for the target container to come online and discover LUNs in the storage system.
2. Create CryptoTarget containers for each target port in the storage array from the Target Container dialog box. Add initiator ports to the container. You must create target containers for those target ports in the configured storage arrays or unassigned target ports before mapping any LUN on the LUN centric view. If you do not create the container, LUN discovery will not function.

NOTE
The controller LUN (LUN 0) must be added to the container as clear text in order for the host to see the LUNs in the container.

For more detailed information on creating a crypto target container, refer to the chapter describing storage arrays in this administrator’s guide.

Remote replication LUNs

The Symmetrix Remote Data Facility (SRDF) transmits data that is being written to both a local Symmetrix array and a remote symmetrix array. The replicated data facilitates a fast switchover to the remote site for data recovery.

SRDF supports the following methods of data replication:

- Synchronous Replication provides real-time mirroring of data between the source Symmetrix and the target Symmetrix systems. Data is written simultaneously to the cache of both systems in real time before the application I/O is completed, thus ensuring the highest possible data availability.
- Semi-Synchronous Replication writes data to the source system, completes the I/O, then synchronizes the data with the target system. Since the I/O is completed prior to synchronizing data with the target system, this method provides an added performance advantage. A second write will not be accepted on a Symmetrix source device until its target device has been synchronized.
- Adaptive Copy Replication transfers data from the source devices to the remote devices without waiting for an acknowledgment. This is especially useful when transferring large amounts of data during data center migrations, consolidations, and in data mobility environments.
- Asynchronous Replication places host writes into chunks and then transfers an entire chunk to the target system. When a complete chunk is received on the target system, the copy cycle is committed. If the SRDF links are lost during data transfer, any partial chunk is discarded, preserving consistency on the target system. This method provides a consistent point-in-time remote image that is not far behind the source system and results in minimal data loss if there is a disaster at the source site.

SRDF pairs

Remote replication is implemented by establishing a synchronized pair of SRDF devices connected by FC or IP links. A local source device is paired with a remote target device while data replication is taking place. While the SRDF devices are paired, the remote target device is not locally accessible for read or write operations. When the data replication operation completes, the pair may be split to enable normal read/write access to both devices. The pair may be restored to restore the data on the local source device.

Figure 54 shows the placement of encryption switches in an SRDF configuration. When encryption is enabled for the primary LUN, encrypted data written by the local application server to the primary LUN is replicated on the secondary LUN. The data is encrypted using a DEK that was generated on the local encryption switch and stored on the local DPM key vault. When each site has an independent key vault, as shown in Figure 54, the key vaults must be synchronized to ensure the availability of the DEK at the remote site. Refer to DPM user documentation for information about how to synchronize the key vaults. Both sites may share the same key vault, which eliminates the need for synchronization across sites. Depending on distance between sites, sharing a key vault may add some latency when retrieving a key.

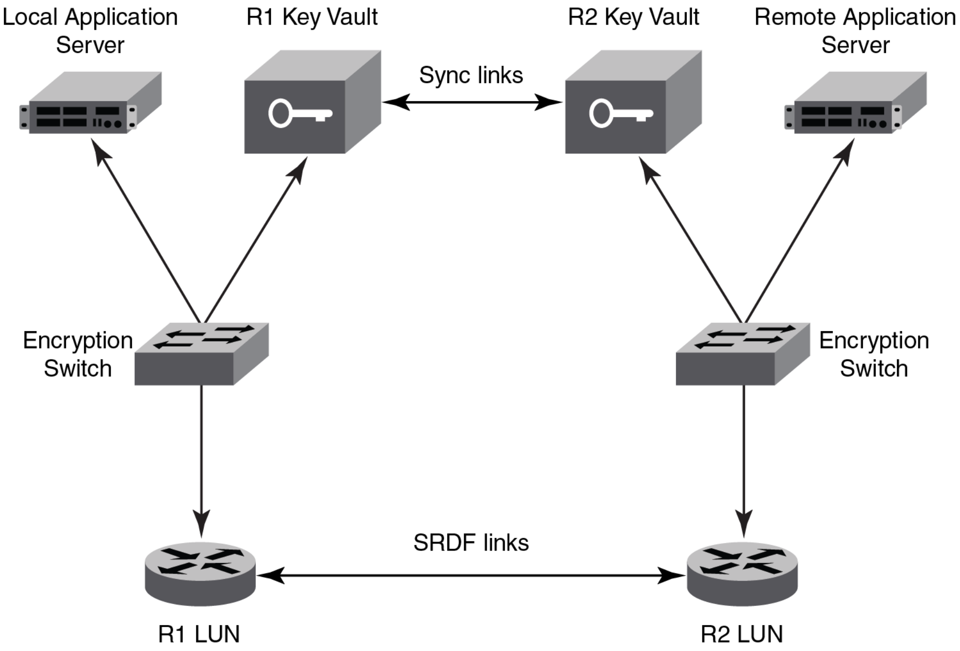


FIGURE 54 Basic SRDF configuration with encryption switches

Metadata requirements and remote replication

When the metadata and key ID are written, the primary metadata on blocks 1–16 is compressed and encrypted. However, there are scenarios whereby these blocks cannot be compressed, and the metadata is not written to the media. If blocks 1–16 are not compressible on the local source device and metadata is not written, obtaining the correct DEK for the remote target device becomes problematic. This problem is avoided by reserving the last three blocks of the LUN for a copy of the metadata. These blocks are not exposed to the host initiator. When a host reads the capacity of the LUN, the size reported is always three blocks less than the actual size. The behavior is enforced by selecting the **New LUN** check box on the **Select LUN** screen of the **Add New Path** wizard when adding LUNs for an SRDF pair (for example, R1 and R2 in Figure 54).

Note the following when using the **New LUN** option:

- Both LUNs that form an SRDF pair must be added to their containers using the **New LUN** option.
- For any site, all paths to a given SRDF device must be configured with the **New LUN** option.
- All LUNs configured with the **New LUN** option will report three blocks less than the actual size when host performs READ CAPACITY 10/READ CAPACITY 16.
- If a LUN is added with the **New LUN** option and with encryption enabled, it will always have valid metadata even if blocks 1–16 of the LUN is not compressible.
- LUNs configured as cleartext must also be added with the **New LUN** option if they are part of an SRDF pair. This is to handle scenarios whereby the LUN policy is changed to encrypted at some later time, and to verify formation of DEK clusters and LUN accessibility prior to enabling encryption for the LUN. When cleartext LUNs are configured with the **New LUN** option, no metadata is written to the last three blocks, but will still report three blocks less than the actual size when host performs READ CAPACITY 10/READ CAPACITY 16.
- The **New LUN** option is used only if a DPM key vault is configured for the encryption group.
- The **New LUN** option can be used only if replication is enabled for the encryption group.
- If the local LUN contains host data, configuring it with the **New LUN** option will cause the data on the last three blocks of the LUN to be lost. Before using the **New LUN** option, you must migrate the contents of the LUN to another LUN that is larger by at least three blocks. The new, larger LUN can then be used when creating the SRDF pair. The remote LUN of the SRDF pair must be of the same size. The original smaller LUN with user data can be decommissioned.

Adding target tape LUNs for encryption

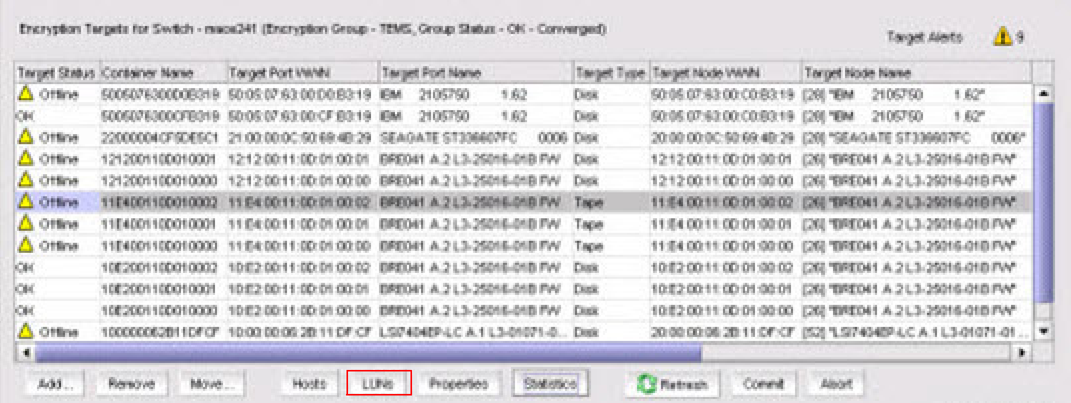
You can configure a Crypto LUN by adding the LUN to the CryptoTarget container and enabling the encryption property on the Crypto LUN. You must add LUNs manually. After you add the LUNs, you must specify the encryption settings.

When configuring a LUN with multiple paths, the same LUN policies must be configured on all paths to the LUN. If there are multiple paths to the same physical LUNs, then the LUNs are added to multiple target containers (one target per storage device port).

1. Select **Configure > Encryption** from the menu task bar to display the **Encryption Center** dialog box (Refer to Figure 6 on page 14).
2. Select a group, switch, or engine from the **Encryption Center Devices** table that contains the storage device to be configured, then select **Group/Switch/Engine > Targets** from the menu task bar.

   NOTE
   You can also select a group, switch, or engine from the **Encryption Center Devices** table, then click the **Targets** icon.

   The **Encryption Targets** dialog box displays (Figure 55). Initially, the table is empty. You must add LUNs manually.



Encryption Targets for Switch - maca241 (Encryption Group - TEMS_Group Status - OK - Converged)

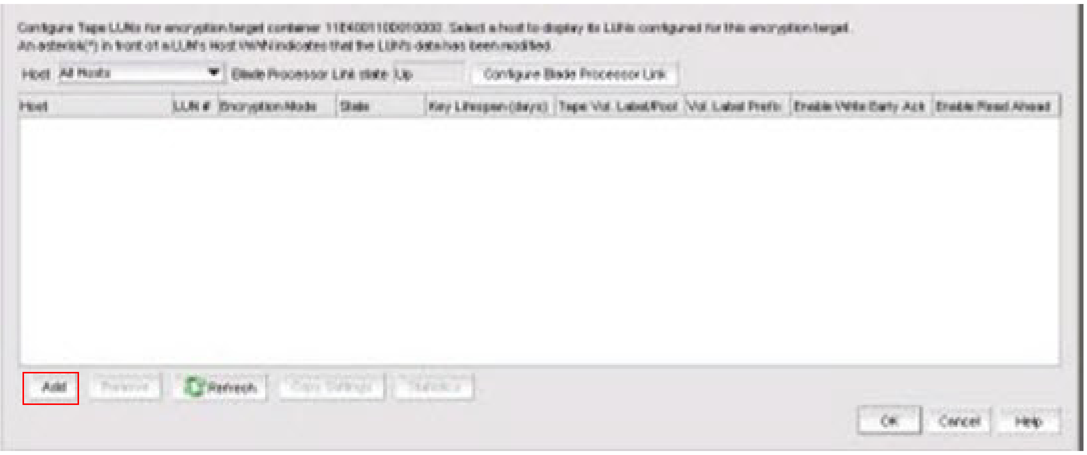
Target Alerts ⚠ 9

Target Status	Container Name	Target Port WWN	Target Port Name	Target Type	Target Node WWN	Target Node Name
⚠ Offline	5005076300D0B319	50:05:07:63:00:D0:B3:19	IBM 2105750 1.62	Disk	50:05:07:63:00:C0:B3:19	[28] "IBM 2105750 1.62"
OK	5005076300CFB319	50:05:07:63:00:CF:B3:19	IBM 2105750 1.62	Disk	50:05:07:63:00:C0:B3:19	[28] "IBM 2105750 1.62"
⚠ Offline	2200000C4CF9DE5C1	21:00:00:0C:50:69:4B:29	SEAGATE ST336607FC 0006	Disk	20:00:00:0C:50:69:4B:29	[28] "SEAGATE ST336607FC 0006"
⚠ Offline	121200110D010001	12:12:00:11:0D:01:00:01	BRE041 A.2 L3-25016-01B FW	Disk	12:12:00:11:0D:01:00:01	[26] "BRE041 A.2 L3-25016-01B FW"
⚠ Offline	121200110D010000	12:12:00:11:0D:01:00:00	BRE041 A.2 L3-25016-01B FW	Disk	12:12:00:11:0D:01:00:00	[26] "BRE041 A.2 L3-25016-01B FW"
⚠ Offline	11E400110D010002	11:E4:00:11:0D:01:00:02	BRE041 A.2 L3-25016-01B FW	Tape	11:E4:00:11:0D:01:00:02	[26] "BRE041 A.2 L3-25016-01B FW"
⚠ Offline	11E400110D010001	11:E4:00:11:0D:01:00:01	BRE041 A.2 L3-25016-01B FW	Tape	11:E4:00:11:0D:01:00:01	[26] "BRE041 A.2 L3-25016-01B FW"
⚠ Offline	11E400110D010000	11:E4:00:11:0D:01:00:00	BRE041 A.2 L3-25016-01B FW	Tape	11:E4:00:11:0D:01:00:00	[26] "BRE041 A.2 L3-25016-01B FW"
OK	10E200110D010002	10:E2:00:11:0D:01:00:02	BRE041 A.2 L3-25016-01B FW	Disk	10:E2:00:11:0D:01:00:02	[26] "BRE041 A.2 L3-25016-01B FW"
OK	10E200110D010001	10:E2:00:11:0D:01:00:01	BRE041 A.2 L3-25016-01B FW	Disk	10:E2:00:11:0D:01:00:01	[26] "BRE041 A.2 L3-25016-01B FW"
OK	10E200110D010000	10:E2:00:11:0D:01:00:00	BRE041 A.2 L3-25016-01B FW	Disk	10:E2:00:11:0D:01:00:00	[26] "BRE041 A.2 L3-25016-01B FW"
⚠ Offline	1000000062B11DFCF	10:00:00:06:2B:11:DF:CF	LS7404EP-LC A.1 L3-01071-0...	Disk	20:00:00:06:2B:11:DF:CF	[52] "LS7404EP-LC A.1 L3-01071-01 ...

FIGURE 55 Encryption Targets dialog box

3. Select a target tape storage device from the **Encryption Targets** table, then click **LUNs**.

   The **Encryption Target Tape LUNs** dialog box displays (Figure 56).



Configure Tape LUNs for encryption target container 11E400110D010000. Select a host to display its LUNs configured for this encryption target.
An asterisk(*) in front of a LUN's Host WWN indicates that the LUN's data has been modified.

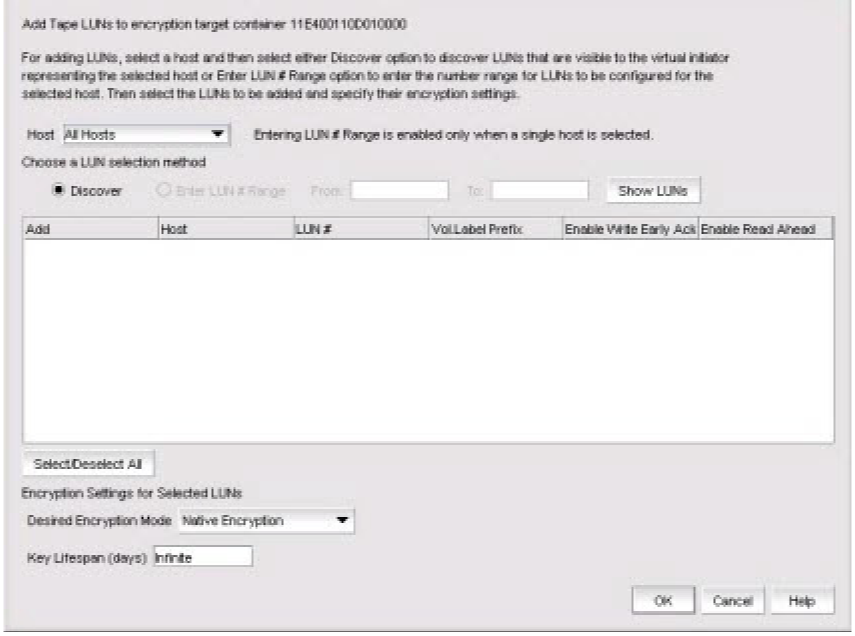
Host: All Hosts   Blade Processor Link state: Up

Host	LUN #	Encryption Mode	State	Key Lifespan (days)	Tape Vol. Label/Pool	Vol. Label Prefix	Enable Write Early Ack	Enable Read Ahead

FIGURE 56 Encryption Target Tape LUNs dialog box

4. Click **Add**.

   The **Add Encryption Target Tape LUNs** dialog box displays (Figure 57). A table of all LUNs in the storage device that are visible to hosts is displayed. LUNs are identified by the **Host** world wide name, **LUN** number, **Volume Label Prefix** number, and **Enable Write Early ACK** and **Enable Read Ahead** status. The LUN numbers may be different for different hosts.



Add Tape LUNs to encryption target container 11E400110D010000

For adding LUNs, select a host and then select either Discover option to discover LUNs that are visible to the virtual initiator representing the selected host or Enter LUN # Range option to enter the number range for LUNs to be configured for the selected host. Then select the LUNs to be added and specify their encryption settings.

Host: All Hosts

Entering LUN # Range is enabled only when a single host is selected.

Choose a LUN selection method

● Discover ○ Enter LUN # Range From: To:

Add	Host	LUN #	Vol.Label Prefix	Enable Write Early Ack	Enable Read Ahead

Encryption Settings for Selected LUNs

Desired Encryption Mode: Native Encryption

Key Lifespan (days): Infinite

FIGURE 57 Add Encryption Target Tape LUNs dialog box

5. Select a host from the **Host** list.

   Before you encrypt a LUN, you must select a host, then either discover LUNs that are visible to the virtual initiator representing the selected host, or enter a range of LUN numbers to be configured for the selected host.

   When you select a specific host, only the LUNs visible to that host are displayed. If you select **All Hosts**, LUNs visible to all configured hosts are displayed. If a LUN is visible to multiple hosts, it is listed once for each host.

6. Choose a LUN to be added to an encryption target container using one of the two following methods:

   - **Discover**: Identifies the exposed logical unit number for a specified initiator. If you already know the exposed LUNs for the various initiators accessing the LUN, you can enter the range of LUNs using the alternative method.
   - **Enter a LUN number range**: Allows you to enter a **From** value and a **To** value to manually enter the logical unit numbers for the selected host(s).

7. Click **Show LUNs**.

   The LUN needed for configuring a Crypto LUN is the LUN that is exposed to a particular initiator.

   The table displays the following information:

   - **Host**: The host on which the LUN is visible.
   - **LUN #**: The logical unit's number.
   - **Vol. Label Prefix**: *Optional*. The user-supplied tape volume label prefix to be included in tape volume labels generated b the switch for encrypted tapes.
   - **Enable Write Early Ack**: When selected, enables tape write pipelining on this tape LUN. Use this option to speed long serial writes to tape, especially for remote backup operations.

- **Enable Read Ahead**: When selected, enables read pre-fetching on this tape LUN. Use this option to speed long serial read operations from tape, especially for remote restore operations.

NOTE
The **Select/Deselect All** button allows you to select or deselect all available LUNs.

8. Select the desired encryption mode. Options are: **Native Encryption**, **DF-Compatible Encryption**, and **Cleartext**.
   - If you change a LUN policy from **Native Encryption** or **DF-Compatible Encryption** to **Clear Text**, you disable encryption.
   - The LUNs of the target that are not enabled for encryption must still be added to the CryptoTarget container with the **Clear Text** encryption mode option.

NOTE
The rekeying interval can only be changed for disk LUNs. For tape LUNs, expiration of the rekeying interval simply triggers the generation of a new key to be used on future tape volumes. Tapes that are already made are not rekeyed. To rekey a tape, you need to read the tape contents using a host application that decrypts the tape contents using the old key, then rewrite the tape, which re-encrypts the data with the new key.

9. Set the **Key Lifespan** setting, then click **OK**.

   The selected tape LUNs are added to the encryption target container.

Moving targets

The **Move Targets** dialog box is used to redistribute which engine encrypts which targets. It is also useful for transferring all targets to another engine before replacing or removing engine hardware. Moving targets to another engine may be done while traffic is flowing between the host and target. Traffic is interrupted for a short time but resumes before the host applications are affected.

1. Select **Configure > Encryption**.

   The **Encryption Center** dialog box displays.

2. Select one or more encryption engines from the **Encryption Center Devices** table, then select **Engine > Targets** from the menu task bar. The encryption engine must be in the same group and same fabric.

   The **Encryption Targets** dialog box displays.

3. Select one or more targets in the Encryption Targets dialog and click **Move**.

   The **Move Targets** dialog box is displayed.

4. Select an encryption engine, then click **OK** to close the dialog and start the move operation.

Configuring encrypted tape storage in a multi-path environment

This example assumes one host is accessing one storage device using two paths:

- The first path is from Host Port A to Target Port A, using Encryption Engine A for encryption.
- The second path is from Host Port B to Target Port B, using Encryption Engine B for encryption.

Encryption Engines A and B are in switches that are already part of Encryption Group *X*.

The following procedure is used to configure this scenario using BNA.

1. Configure Host Port A and Target Port A in the same zone by selecting **Configure** > **Zoning** from BNA's main menu.
2. Configure Host Port B and Target Port B in the same zone by selecting **Configure** > **Zoning** from BNA's main menu.
3. Select **Configure** > **Encryption** from the menu task bar to display the **Encryption Center** dialog box (Refer to Figure 6 on page 14).
4. Click **View Groups** to display the encryption groups if groups are not already displayed.
5. Select Encryption Group *X*, then click the **Targets** icon.
6. From the **Encryption Targets** dialog box, click **Add** to open the **Configure Storage Encryption** wizard. Use the wizard to create a target container for Encryption Engine A with Target Port A and Host Port A.
7. Repeat Step 6 to create a target container for Encryption Engine B with Target Port B and Host Port B.

   Up to this point, BNA has been automatically committing changes as they are made. The targets and hosts are now fully configured; only the LUN configuration remains.
8. In the **Encryption Targets** dialog box, select Target Port A, click **LUNs**, then click **Add**. Select the LUNs to be encrypted and the encryption policies for the LUNs.
9. In the **Encryption Targets** dialog box, select Target Port B, click **LUNs**, then click **Add**. Select the LUNs to be encrypted and the encryption policies for the LUNs, making sure that the encryption policies match the policies specified in the other path.
10. Click **Commit** to make the LUN configuration changes effective in both paths simultaneously.

BNA does not automatically commit LUN configuration changes. You must manually commit any LUN configuration changes, even in non-multipath environments. Committing LUN configuration changes manually allows the matching changes made in a multi-path environment to be committed together, preventing cases where one path may be encrypting and another path is not, thus causing corrupted data.

NOTE
There is a limit of 16 uncommitted tape LUN configuration changes. When adding more than eight LUNs in a multi-path environment, repeat step 8 and step 9 above, adding only eight LUNs to each target container at a time. Each commit operation will commit 16 LUNs, eight in each path.

Tape LUN write early and read ahead

The tape LUN write early and read ahead feature uses tape pipelining and prefetch to speed serial access to tape storage. These features are particularly useful when performing backup and restore operations, especially over long distances.

You can enable tape LUN write early and read ahead while adding the tape LUN for encryption, or you can enable or disable these features after the tape LUN has been added for encryption.

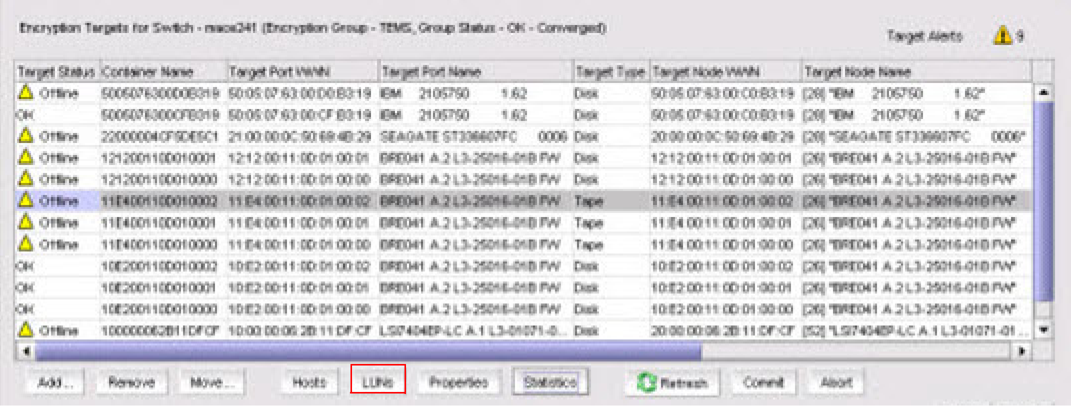
Enabling and disabling tape LUN write early and read ahead

To enable or disable tape LUN write early and read ahead, follow these steps:

1. Select **Configure > Encryption** from the menu task bar to display the **Encryption Center** dialog box (Refer to Figure 6 on page 14).
2. Select a group, switch, or engine from the **Encryption Center Devices** table, then select **Group/Switch/Engine > Targets** from the menu task bar.

   NOTE
   You can also select a group, switch, or engine from the **Encryption Center Devices** table, then click the **Targets** icon.

   The **Encryption Targets** dialog box displays (Figure 58).



Encryption Targets for Switch - maca241 (Encryption Group - TEMS_Group Status - OK - Converged)

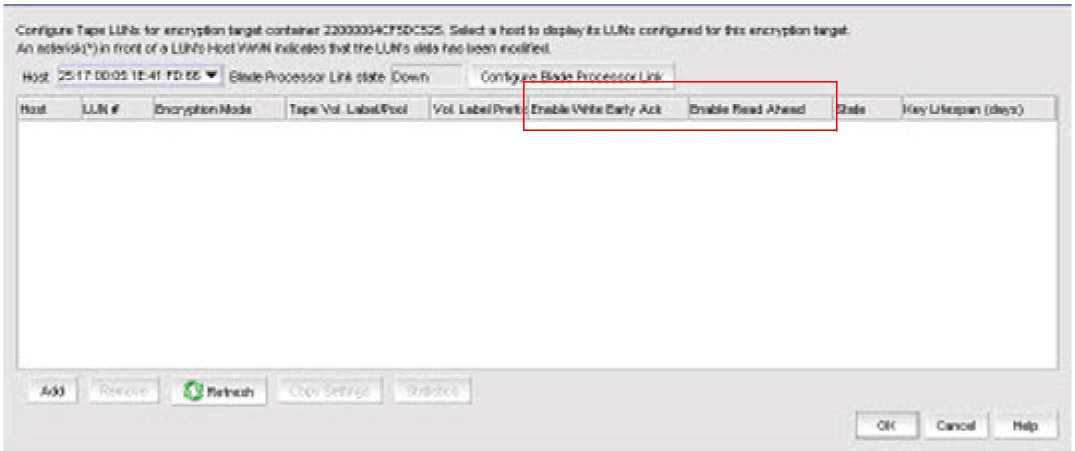
Target Alerts ⚠ 9

Target Status	Container Name	Target Port WWN	Target Port Name	Target Type	Target Node WWN	Target Node Name
⚠ Offline	5005076300D0B319	50:05:07:63:00:D0:B3:19	IBM 2105750 1.62	Disk	50:05:07:63:00:C0:B3:19	[28] "IBM 2105750 1.62"
OK	5005076300CFB319	50:05:07:63:00:CF:B3:19	IBM 2105750 1.62	Disk	50:05:07:63:00:C0:B3:19	[28] "IBM 2105750 1.62"
⚠ Offline	22000004CF9DE5C1	21:00:00:0C:50:69:4B:29	SEAGATE ST336607FC 0006	Disk	20:00:00:0C:50:69:4B:29	[28] "SEAGATE ST336607FC 0006"
⚠ Offline	1212001100010001	12:12:00:11:00:01:00:01	BRE041 A.2 L3-25016-01B FW	Disk	12:12:00:11:00:01:00:01	[26] "BRE041 A.2 L3-25016-01B FW"
⚠ Offline	1212001100010000	12:12:00:11:00:01:00:00	BRE041 A.2 L3-25016-01B FW	Disk	12:12:00:11:00:01:00:00	[26] "BRE041 A.2 L3-25016-01B FW"
⚠ Offline	11E4001100010002	11:E4:00:11:00:01:00:02	BRE041 A.2 L3-25016-01B FW	Tape	11:E4:00:11:00:01:00:02	[26] "BRE041 A.2 L3-25016-01B FW"
⚠ Offline	11E4001100010001	11:E4:00:11:00:01:00:01	BRE041 A.2 L3-25016-01B FW	Tape	11:E4:00:11:00:01:00:01	[26] "BRE041 A.2 L3-25016-01B FW"
⚠ Offline	11E4001100010000	11:E4:00:11:00:01:00:00	BRE041 A.2 L3-25016-01B FW	Tape	11:E4:00:11:00:01:00:00	[26] "BRE041 A.2 L3-25016-01B FW"
OK	10E2001100010002	10:E2:00:11:00:01:00:02	BRE041 A.2 L3-25016-01B FW	Disk	10:E2:00:11:00:01:00:02	[26] "BRE041 A.2 L3-25016-01B FW"
OK	10E2001100010001	10:E2:00:11:00:01:00:01	BRE041 A.2 L3-25016-01B FW	Disk	10:E2:00:11:00:01:00:01	[26] "BRE041 A.2 L3-25016-01B FW"
OK	10E2001100010000	10:E2:00:11:00:01:00:00	BRE041 A.2 L3-25016-01B FW	Disk	10:E2:00:11:00:01:00:00	[26] "BRE041 A.2 L3-25016-01B FW"
⚠ Offline	100000062B11DFCF	10:00:00:06:2B:11:DF:CF	LSI7404EP-LC A.1 L3-01071-0...	Disk	20:00:00:06:2B:11:DF:CF	[52] "LSI7404EP-LC A.1 L3-01071-01 ...

FIGURE 58 Encryption Targets dialog box

3. Select a target tape storage device from the table, then click **LUNs**.

   The **Encryption Target Tape LUNs** dialog box displays (Figure 59).



Configure Tape LUNs for encryption target container 22000004CF5DC525. Select a host to display its LUNs configured for this encryption target.
An asterisk(*) in front of a LUN's Host WWN indicates that the LUN's data has been modified.

Host 25:17:00:05:1E:41:FD:E6

Blade Processor Link state Down

FIGURE 59 Encryption Target Tape LUNs dialog box - Setting tape LUN read ahead and write early

4. In the **Enable Write EarlyAck** and **Enable Read Ahead** columns, when the table is populated, you can set these features as desired for each LUN:
   - To enable write early for a specific tape LUN, select **Enable Write Early Ack** for that LUN.
   - To enable read ahead for a specific LUN, select **Enable Read Ahead** for that LUN.
   - To disable write early for a specific tape LUN, deselect **Enable Write Early Ack** for that LUN.
   - To disable read ahead for a specific LUN, deselect **Enable Read Ahead** for that LUN.
5. Click **OK.**
6. Commit the changes on the related crypto target container:

   a. Select **Configure** > **Encryption** from the menu task bar to display the **Encryption Center** dialog box (Refer to Figure 6 on page 14).

   b. Select a group, switch, or engine from the **Encryption Center Devices** table that contains the storage device to be configured, then select **Group/Switch/Engine** > **Targets** from the menu task bar.

   NOTE
   You can also select a group, switch, or engine from the **Encryption Center Devices** table, then click the **Targets** icon.

   c. Select the appropriate crypto target container, then click **Commit.**

Tape LUN statistics

This feature enables you to view and clear statistics for tape LUNs. These statistics include the number of compressed blocks, uncompressed blocks, compressed bytes and uncompressed bytes written to a tape LUN.

The tape LUN statistics are cumulative and change as the host writes more data on tape. You can clear the statistics to monitor compression ratio of ongoing host I/Os.

The encryption management application allows you to select tape LUN from either a tape LUN container through the **Encryption Targets** dialog box, or from the **Target Tape LUNs** dialog box.

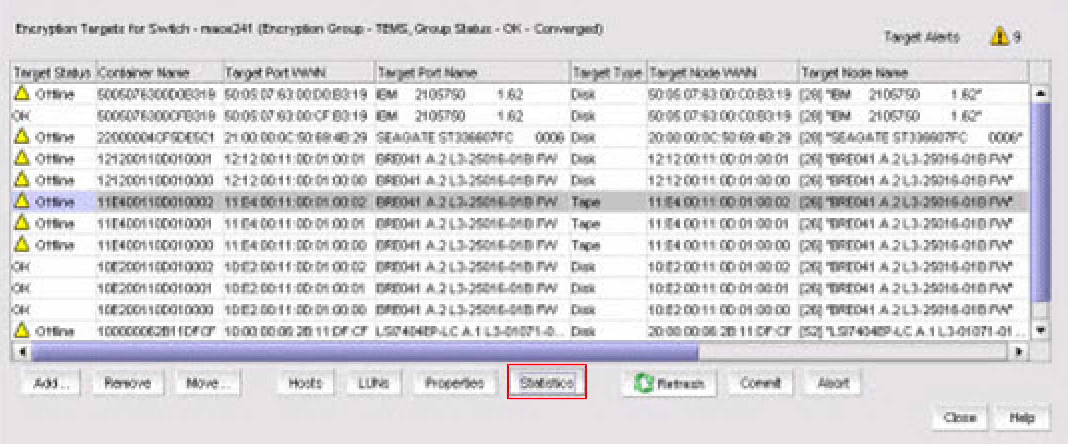
Viewing and clearing tape container statistics

You can view LUN statistics for an entire crypto tape container or for specific LUNs.

To view or clear statistics for tape LUNs in a container, follow these steps:

1. Select **Configure > Encryption** from the menu task bar to display the **Encryption Center** dialog box (Refer to Figure 6 on page 14).
2. Select a group from the **Encryption Center Devices** table, then select **Group > Targets** from the menu task bar.

   The **Encryption Targets** dialog box displays (Figure 60). A list of the configured crypto target containers is displayed.



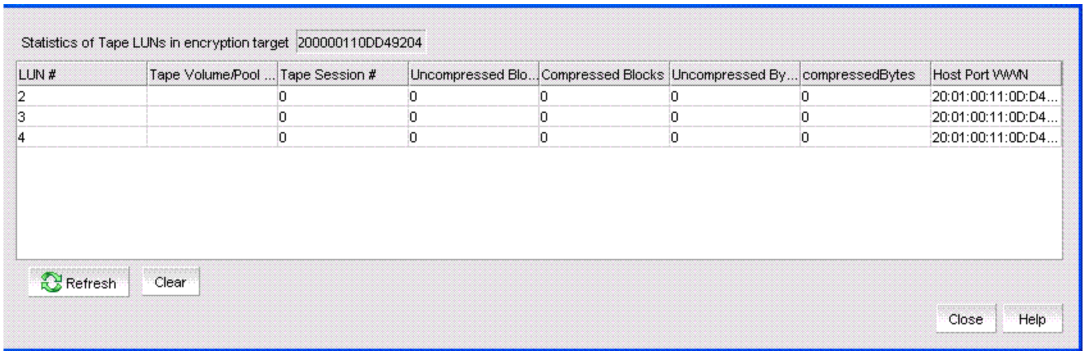
Encryption Targets for Switch - maca241 (Encryption Group - TEMS, Group Status - OK - Converged)

Target Status	Container Name	Target Port WWN	Target Port Name	Target Type	Target Node WWN	Target Node Name
⚠ Offline	5005076300D0B319	50:05:07:63:00:D0:B3:19	IBM 2105750 1.62	Disk	50:05:07:63:00:C0:B3:19	[28] "IBM 2105750 1.62"
OK	5005076300CFB319	50:05:07:63:00:CF:B3:19	IBM 2105750 1.62	Disk	50:05:07:63:00:C0:B3:19	[28] "IBM 2105750 1.62"
⚠ Offline	22000004CF9DE5C1	21:00:00:0C:50:69:4B:29	SEAGATE ST336607FC 0006	Disk	20:00:00:0C:50:69:4B:29	[28] "SEAGATE ST336607FC 0006"
⚠ Offline	121200110D010001	12:12:00:11:0D:01:00:01	BRE041 A.2 L3-25016-01B FW	Disk	12:12:00:11:0D:01:00:01	[26] "BRE041 A.2 L3-25016-01B FW"
⚠ Offline	121200110D010000	12:12:00:11:0D:01:00:00	BRE041 A.2 L3-25016-01B FW	Disk	12:12:00:11:0D:01:00:00	[26] "BRE041 A.2 L3-25016-01B FW"
⚠ Offline	11E400110D010002	11:E4:00:11:0D:01:00:02	BRE041 A.2 L3-25016-01B FW	Tape	11:E4:00:11:0D:01:00:02	[26] "BRE041 A.2 L3-25016-01B FW"
⚠ Offline	11E400110D010001	11:E4:00:11:0D:01:00:01	BRE041 A.2 L3-25016-01B FW	Tape	11:E4:00:11:0D:01:00:01	[26] "BRE041 A.2 L3-25016-01B FW"
⚠ Offline	11E400110D010000	11:E4:00:11:0D:01:00:00	BRE041 A.2 L3-25016-01B FW	Tape	11:E4:00:11:0D:01:00:00	[26] "BRE041 A.2 L3-25016-01B FW"
OK	10E200110D010002	10:E2:00:11:0D:01:00:02	BRE041 A.2 L3-25016-01B FW	Disk	10:E2:00:11:0D:01:00:02	[26] "BRE041 A.2 L3-25016-01B FW"
OK	10E200110D010001	10:E2:00:11:0D:01:00:01	BRE041 A.2 L3-25016-01B FW	Disk	10:E2:00:11:0D:01:00:01	[26] "BRE041 A.2 L3-25016-01B FW"
OK	10E200110D010000	10:E2:00:11:0D:01:00:00	BRE041 A.2 L3-25016-01B FW	Disk	10:E2:00:11:0D:01:00:00	[26] "BRE041 A.2 L3-25016-01B FW"
⚠ Offline	100000062B11DFCF	10:00:00:06:2B:11:DF:CF	LSI7404EP-LC A.1 L3-01071-0...	Disk	20:00:00:06:2B:11:DF:CF	[52] "LSI7404EP-LC A.1 L3-01071-01 ...

FIGURE 60 Encryption Targets dialog box

3. Select **Tape** as the container of type for which to display or clear statistics, then click **Statistics**.

   The **Tape LUN Statistics** dialog box displays (Figure 61). A list of the statistics for all LUNs that are members of the selected tape container is displayed.



Statistics of Tape LUNs in encryption target 200000110DD49204

LUN #	Tape Volume/Pool ...	Tape Session #	Uncompressed Blo...	Compressed Blocks	Uncompressed By...	compressedBytes	Host Port WWN
2		0	0	0	0	0	20:01:00:11:0D:D4...
3		0	0	0	0	0	20:01:00:11:0D:D4...
4		0	0	0	0	0	20:01:00:11:0D:D4...

FIGURE 61 Tape LUN Statistics dialog box

The dialog box contains the following information:

- **LUN #:** The number of the logical unit for which statics are displayed.
- **Tape Volume/Pool:** The tape volume label of the currently-mounted tape, if a tape session is currently in progress.
- **Tape Session #:** The number of the ongoing tape session.

- **Uncompressed blocks:** The number of uncompressed blocks written to tape.
- **Compressed blocks:** The number of compressed blocks written to tape.
- **Uncompressed Bytes:** The number of uncompressed bytes written to tape.
- **Compressed Bytes:** The number of compressed bytes written to tape.
- **Host Port WWN:** The WWN of the host port that is being used for the write operation.
- A **Refresh** button updates the statistics on the display since the last reset.
- A **Clear** button resets all statistics in the display.

4. To clear the tape LUN statistics for all member LUNs for the container, click **Clear**, then click **Yes** to confirm.

To view statistics for specific LUNs:

1. Select a tape container, then click **LUNs**.
2. From the **Target Tape LUNs** dialog box, select the LUNs you want to monitor.

Viewing and clearing tape LUN statistics for specific tape LUNs

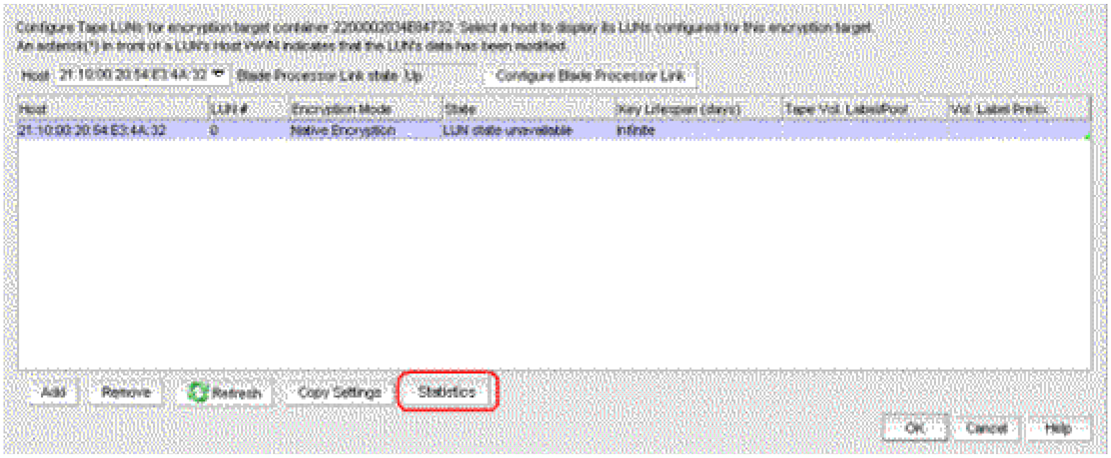
To view or clear statistics for tape LUNs in a container, complete these steps:

1. Select **Configure** > **Encryption** from the menu task bar to display the **Encryption Center** dialog box (Refer to Figure 6 on page 14).
2. Select a group, switch, or engine from the **Encryption Center Devices** table that contains the storage device to be configured, then select **Group/Switch/Engine** > **Targets** from the menu task bar.

   NOTE
   You can also select a group, switch, or engine from the **Encryption Center Devices** table, then click the **Targets** icon.

   The **Encryption Targets** dialog box displays (Figure 58).
3. Select a tape target storage device, then click **LUNs**.

   The **Target Tape LUNs** dialog box displays (Figure 62). A list of the configured tape LUNs is displayed.



Configure Tape LUNs for encryption target container 2200002004E04732. Select a host to display its LUNs configured for this encryption target.
An asterisk(*) in front of a LUN's Host WWN indicates that the LUN's data has been modified.

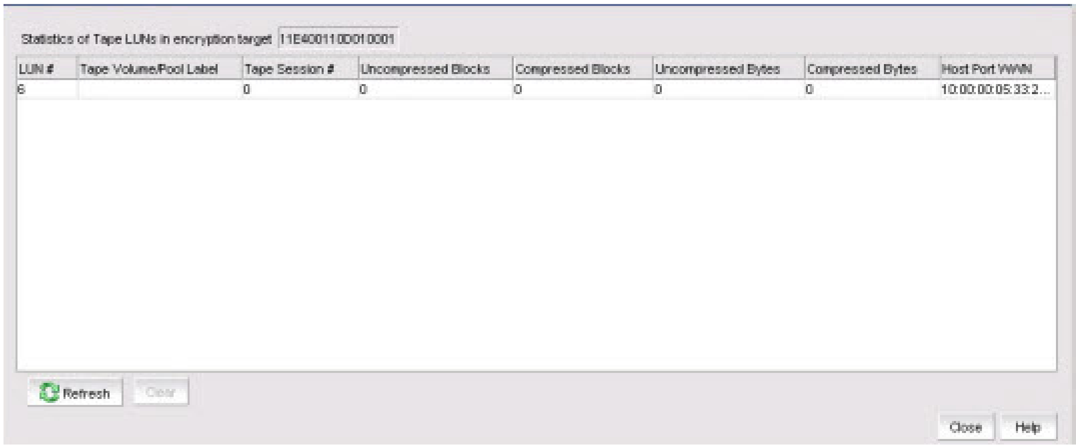
Host: 21:10:00:20:54:E3:4A:32 Blade Processor Link state: Up

Host	LUN #	Encryption Mode	State	Key Lifespan (days)	Tape Vol. Label/Pool	Vol. Label Prefix
21:10:00:20:54:E3:4A:32	0	Native Encryption	LUN state unavailable	Infinite		

FIGURE 62 Target Tape LUNs dialog box

4. Select the LUN or LUNs for which to display or clear statistics, then click **Statistics**.

   The **Tape LUN Statistics** dialog box displays (Figure 63). The statistic results based on the LUN or LUNs you selected is displayed. Tape LUN statistics are cumulative.



Statistics of Tape LUNs in encryption target 11E40011 0D01 0001

LUN #	Tape Volume/Pool Label	Tape Session #	Uncompressed Blocks	Compressed Blocks	Uncompressed Bytes	Compressed Bytes	Host Port WWN
6		0	0	0	0	0	10:00:00:05:33:2...

FIGURE 63 Tape LUN Statistics dialog box

The dialog box contains the following information:

- **LUN #:** The number of the logical unit for which statics are displayed.
- **Tape Volume/Pool:** The tape volume label of the currently-mounted tape, if a tape session is currently in progress.
- **Tape Session #:** The number of the ongoing tape session.
- **Uncompressed blocks:** The number of uncompressed blocks written to tape.
- **Compressed blocks:** The number of compressed blocks written to tape.
- **Uncompressed Bytes:** The number of uncompressed bytes written to tape.
- **Compressed Bytes:** The number of compressed bytes written to tape.
- **Host Port WWN:** The WWN of the host port that is being used for the write operation.
- A **Refresh** button updates the statistics on the display since the last reset.
- A **Clear** button resets all statistics in the display.

5. Do either of the following:

   - Click **Clear** to clear the tape LUN statistics, then click **Yes** to confirm.
   - Click **Refresh** to view the current statistics cumulative since the last reset.

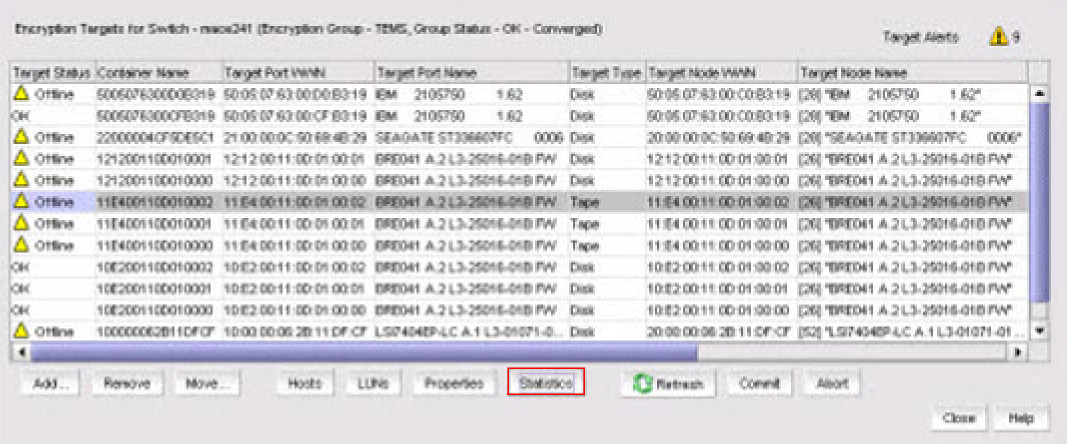
Viewing and clearing statistics for tape LUNs in a container

To view or clear statistics for tape LUNs in a container, follow these steps:

1. Select **Configure > Encryption** from the menu task bar to display the **Encryption Center** dialog box (Refer to Figure 6 on page 14).
2. Select a group, switch, or engine from the **Encryption Center Devices** table that contains the storage device to be configured, then select **Group/Switch/Engine > Targets** from the menu task bar.

NOTE
You can also select a group, switch, or engine from the **Encryption Center Devices** table, then click the **Targets** icon.

The **Encryption Targets** dialog box displays (Figure 64). A list of configured crypto target containers is displayed.



Encryption Targets for Switch - mace241 (Encryption Group - TEMS_Group Status - OK - Converged)

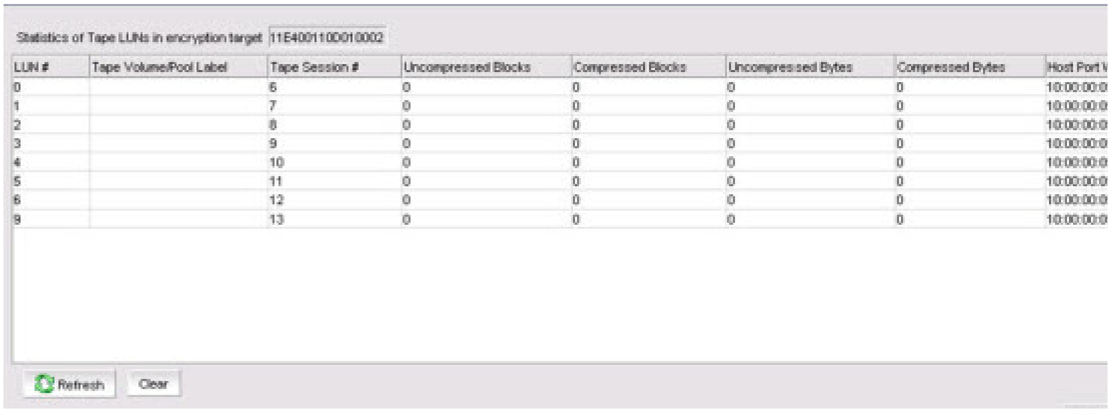
Target Alerts ⚠ 9

Target Status	Container Name	Target Port WWN	Target Port Name	Target Type	Target Node WWN	Target Node Name
⚠ Offline	5005076300D0B319	50:05:07:63:00:D0:B3:19	IBM 2105750 1.62	Disk	50:05:07:63:00:C0:B3:19	[28] "IBM 2105750 1.62"
OK	5005076300CFB319	50:05:07:63:00:CF:B3:19	IBM 2105750 1.62	Disk	50:05:07:63:00:C0:B3:19	[28] "IBM 2105750 1.62"
⚠ Offline	22000004CF9DE5C1	21:00:00:0C:50:69:4B:29	SEAGATE ST336607FC 0006	Disk	20:00:00:0C:50:69:4B:29	[28] "SEAGATE ST336607FC 0006"
⚠ Offline	121200110D010001	12:12:00:11:0D:01:00:01	BRE041 A.2 L3-25016-01B FW	Disk	12:12:00:11:0D:01:00:01	[26] "BRE041 A.2 L3-25016-01B FW"
⚠ Offline	121200110D010000	12:12:00:11:0D:01:00:00	BRE041 A.2 L3-25016-01B FW	Disk	12:12:00:11:0D:01:00:00	[26] "BRE041 A.2 L3-25016-01B FW"
⚠ Offline	11E400110D010002	11:E4:00:11:0D:01:00:02	BRE041 A.2 L3-25016-01B FW	Tape	11:E4:00:11:0D:01:00:02	[26] "BRE041 A.2 L3-25016-01B FW"
⚠ Offline	11E400110D010001	11:E4:00:11:0D:01:00:01	BRE041 A.2 L3-25016-01B FW	Tape	11:E4:00:11:0D:01:00:01	[26] "BRE041 A.2 L3-25016-01B FW"
⚠ Offline	11E400110D010000	11:E4:00:11:0D:01:00:00	BRE041 A.2 L3-25016-01B FW	Tape	11:E4:00:11:0D:01:00:00	[26] "BRE041 A.2 L3-25016-01B FW"
OK	10E200110D010002	10:E2:00:11:0D:01:00:02	BRE041 A.2 L3-25016-01B FW	Disk	10:E2:00:11:0D:01:00:02	[26] "BRE041 A.2 L3-25016-01B FW"
OK	10E200110D010001	10:E2:00:11:0D:01:00:01	BRE041 A.2 L3-25016-01B FW	Disk	10:E2:00:11:0D:01:00:01	[26] "BRE041 A.2 L3-25016-01B FW"
OK	10E200110D010000	10:E2:00:11:0D:01:00:00	BRE041 A.2 L3-25016-01B FW	Disk	10:E2:00:11:0D:01:00:00	[26] "BRE041 A.2 L3-25016-01B FW"
⚠ Offline	100000062B11DFCF	10:00:00:06:2B:11:DF:CF	LSI7404EP-LC A.1 L3-01071-0...	Disk	20:00:00:06:2B:11:DF:CF	[52] "LSI7404EP-LC A.1 L3-01071-01...

FIGURE 64 Encryption Targets dialog box

3. Select **Tape as** the container of type for which to display or clear statistics, then click **Statistics**.

   The **Tape LUN Statistics** dialog box displays (Figure 65). The statistics for all LUNs that are members of the selected tape container are displayed.



Statistics of Tape LUNs in encryption target 11E400110D010002

LUN #	Tape Volume/Pool Label	Tape Session #	Uncompressed Blocks	Compressed Blocks	Uncompressed Bytes	Compressed Bytes	Host Port V
0		6	0	0	0	0	10:00:00:0
1		7	0	0	0	0	10:00:00:0
2		8	0	0	0	0	10:00:00:0
3		9	0	0	0	0	10:00:00:0
4		10	0	0	0	0	10:00:00:0
5		11	0	0	0	0	10:00:00:0
6		12	0	0	0	0	10:00:00:0
9		13	0	0	0	0	10:00:00:0

FIGURE 65 Tape LUN Statistics dialog box

The dialog box contains the following information:

- **LUN #:** The number of the logical unit for which statics are displayed.
- **Tape Volume/Pool:** The tape volume label of the currently-mounted tape, if a tape session is currently in progress.
- **Tape Session #:** The number of the ongoing tape session.
- **Uncompressed blocks:** The number of uncompressed blocks written to tape.
- **Compressed blocks:** The number of compressed blocks written to tape.

- **Uncompressed Bytes:** The number of uncompressed bytes written to tape.
- **Compressed Bytes:** The number of compressed bytes written to tape.
- **Host Port WWN:** The WWN of the host port that is being used for the write operation.

4. Do either of the following:
   - Click **Clear** to clear the tape LUN statistics for member LUNs in the container, then click **Yes** to confirm.
   - Click **Refresh** to update the tape LUN statistics on the display.

Encryption engine rebalancing

If you are currently using encryption and running Fabric OS 6.3.x or earlier, you are hosting tape and disk target containers on different encryption switches or blades. Beginning with Fabric OS 6.4, disk and tape target containers can be hosted on the same switch or blade. Hosting both disk and tape target containers on the same switch or blade might result in a drop in throughput, but it can reduce cost by reducing the number of switches or blades needed to support encrypted I/O in environments that use both disk and tape.

The throughput drop can be mitigated by rebalancing the tape and disk target containers across the encryption engine. This ensures that the tape and disk target containers are distributed within the encryption engine for maximum throughput.

All nodes within an encryption group must be upgraded to Fabric OS 6.4 or later to support hosting disk and tape target containers on the same encryption engine. If any node within an encryption group is running an earlier release, disk and tape containers must continue to be hosted on separate encryption engines.

During rebalancing operations, be aware of the following:

- You might notice a slight disruption in Disk I/O. In some cases, manual intervention may be needed.
- Backup jobs to tapes might need to be restarted after rebalancing is completed.

To determine if rebalancing is recommended for an encryption engine, check the encryption engine properties. Beginning with Fabric OS 6.4, a field is added that indicates whether or not rebalancing is recommended.

You might be prompted to rebalance during the following operations:

- When adding a new disk or tape target container.
- When removing an existing disk or tape target container.
- After failover to a backup encryption engine in an HA cluster.
- After a failed encryption engine in an HA cluster is recovered, and failback processing has occurred.

Rebalancing an encryption engine

To rebalance an encryption engine, complete the following steps:

1. Select **Configure > Encryption** from the menu task bar to display the **Encryption Center** dialog box (Refer to Figure 6 on page 14).
2. Select an engine, then select **Engine > Re-Balance** from the menu task bar.

   A warning message displays, noting the potential disruption of disk and tape I/O, and that the operation may take several minutes.
3. Click **Yes** to begin rebalancing.

Master keys

Master keys belong to the group and are managed from **Group Properties**.

When an opaque key vault is used, a master key is used to encrypt the data encryption keys. The master key status indicates whether a master key is used and whether it has been backed up. Encryption is not allowed until the master key has been backed up.

Only the active master key can be backed up, and multiple backups are recommended. You can back up or restore the master key to the key vault, to a file, or to a recovery card set. A recovery card set is set of smart cards. Each recovery card holds a portion of the master key. The cards must be gathered and read together from a card reader attached to a PC running BNA to restore the master key.

Although it is generally not necessary to create a new master key, you might be required to create one due to the following:

- The previous master key has been compromised.
- Corporate policy might require a new master key every year for security purposes.

When you create a new master key, the former active master key automatically becomes the alternate master key.

The new master key cannot be used (no new data encryption keys can be created, so no new encrypted LUNs can be configured), until you back up the new master key. After you have backed up the new master key, it is strongly recommended that all encrypted disk LUNs be rekeyed. rekeying causes a new data encryption key to be created and encrypted using the new active master key, thereby removing any dependency on the old master key. Refer to "Creating a master key" on page 93 for more information.

Master key actions are disabled if they are unavailable. For example:

- The user does not have Storage Encryption Security permissions.
- The group leader is not discovered or managed by BNA.

NOTE
It is important to back up the master key because if the master key is lost, none of the data encryption keys can be restored and none of the encrypted data can be decrypted.

Active master key

The active master key is used to encrypt newly created data encryption keys (DEKs) prior to sending them to a key vault to be stored. You can restore the active master key under the following conditions:

- The active master key has been lost, which happens if all encryption engines in the group have been zeroized or replaced with new hardware at the same time.
- You want multiple encryption groups to share the same active master key. Groups should share the same master key if the groups share the same key vault and if tapes (or disks) are going to be exchanged regularly between the groups.

Alternate master key

The alternate master key is used to decrypt data encryption keys that were not encrypted with the active master key. Restore the alternate master key for the following reasons:

- To read an old tape that was created when the group used a different active master key.
- To read a tape (or disk) from a different encryption group that uses a different active master key.

Master key actions

NOTE
Master keys belong to the group and are managed from Group Properties.

Master key actions are as follows:

- **Backup master key**: Enabled any time a master key exists. Selecting this option launches the **Backup Master Key for Encryption Group** dialog box.

  You can back up the master key to a file, to a key vault, or to a smart card. You can back up the master key multiple times to any of these media in case you forget the passphrase you originally used to back up the master key, or if multiple administrators each needs a passphrase for recovery. Refer to the following procedures for more information:

  - "Saving the master key to a file" on page 87
  - "Saving a master key to a key vault" on page 88
  - "Saving a master key to a smart card set" on page 89

  You must back up the master key when the status is **Created but not backed up.**

- **Restore master key**: Enabled when no master key exists or the previous master key has been backed up. This option is also enabled when using a DPM key vault.

  - When this option is selected, the **Restore Master Key for Encryption Group** dialog box displays, from which you can restore a master key from a file, key vault, or smart card set. Refer to the following procedures for more information:

  - "Restoring a master key from a file" on page 90
  - "Restoring a master key from a key vault" on page 91
  - "Restoring a master key from a smart card set" on page 92

- **Create new master key**: Enabled when no master key exists, or the previous master key has been backed up. Refer to "Creating a master key" on page 93.

  You must create a new master key when the status is **Required but not created.**

  NOTE
  If a master key was not created, **Not Used** is displayed as the status and the **Master Key Actions** list is grayed out. In this case, you must create a new master key. Additional master key statuses are: **Backed up but not propagated** and **Created and backed up.**

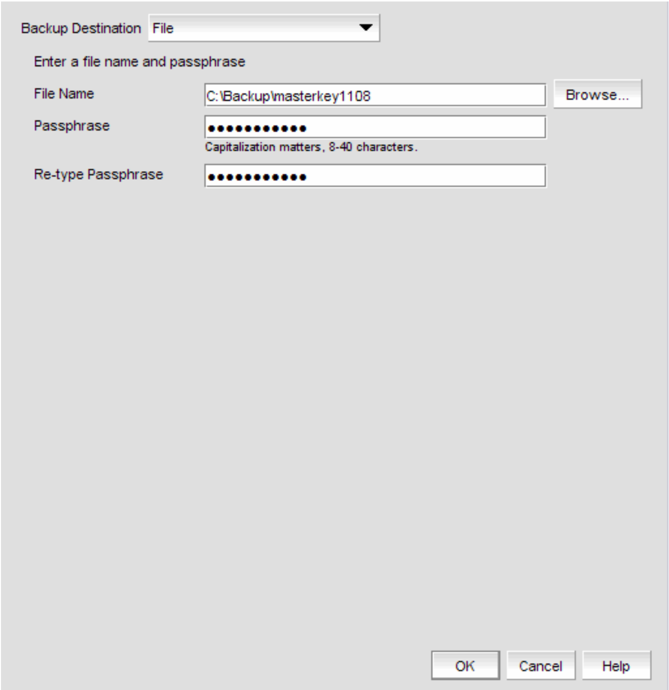
Saving the master key to a file

Use the following procedure to save the master key to a file.

1. Select **Configure > Encryption** from the menu task bar to display the **Encryption Center** dialog box (Refer to Figure 6 on page 14).
2. Select a group from the **Encryption Center Devices** table, then select **Group > Security** from the menu task bar.

   The **Encryption Group Properties** dialog box displays with the **Security** tab selected.
3. Select **Backup Master Key** as the **Master Key Action**.

   The **Master Key Backup** dialog box displays (Figure 66), but only if the master key has already been generated.



Backup Destination: File

Enter a file name and passphrase

File Name: C:\Backup\masterkey1108

Passphrase: ●●●●●●●●●●●●

Capitalization matters, 8-40 characters.

Re-type Passphrase: ●●●●●●●●●●●●

FIGURE 66 Backup Destination (to file) dialog box

4. Select **File** as the **Backup Destination.**
5. Enter a file name, or browse to the desired location.

6. Enter the passphrase, which is required for restoring the master key. The passphrase can be between eight and 40 characters, and any character is allowed.

7. Re-enter the passphrase for verification, then click **OK**.

ATTENTION
Save the passphrase. This passphrase is required if you ever need to restore the master key from the file.

Saving a master key to a key vault

Use the following procedure to save the master key to a key vault.

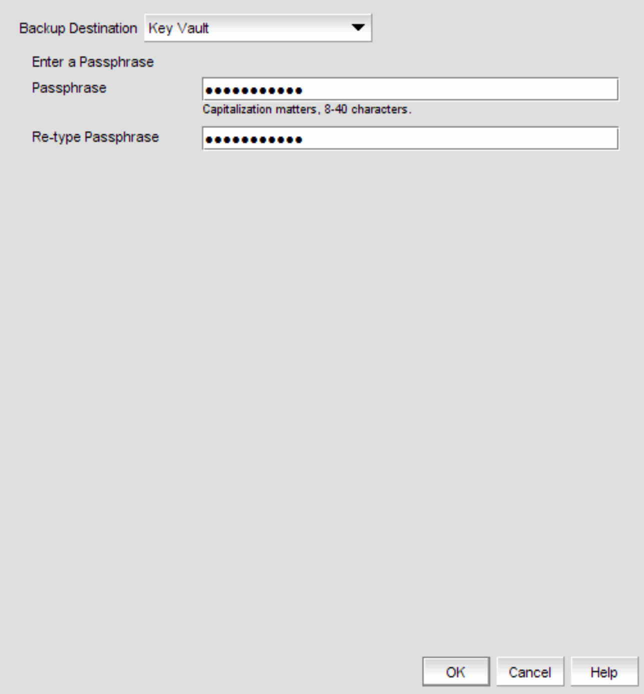
1. Select **Configure > Encryption** from the menu task bar to display the **Encryption Center** dialog box (Refer to Figure 6 on page 14).

2. Select a group from the **Encryption Center Devices** table, then select **Group > Security** from the menu task bar.

   The **Encryption Group Properties** dialog box displays with the **Security** tab selected.

3. Select **Backup Master Key** as the **Master Key Action**.

   The **Backup Master Key for Encryption Group** dialog box displays (Figure 67).


Backup Destination: Key Vault

Enter a Passphrase

Passphrase: ●●●●●●●●●●●

Capitalization matters, 8-40 characters.

Re-type Passphrase: ●●●●●●●●●●●

FIGURE 67 Backup Destination (to key vault) dialog box

4. Select **Key Vault** as the **Backup Destination**.

5. Enter the passphrase, which is required for restoring the master key. The passphrase can be between eight and 40 characters, and any character is allowed.

6. Re-enter the passphrase for verification, then click **OK.**

   A dialog box displays that shows the **Key ID**. The Key ID identifies the storage location in the key vault.

7. Store both the Key ID and the passphrase in a secure place. Both will be required to restore the master key in the future.

8. Click **OK.** after you have copied the **Key ID**.

Saving a master key to a smart card set

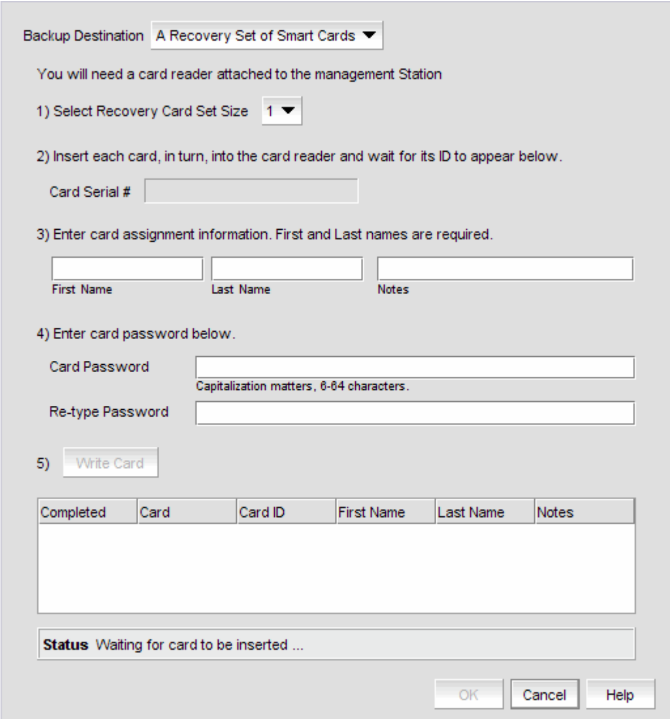
1. Select **Configure** > **Encryption** from the menu task bar to display the **Encryption Center** dialog box (Refer to Figure 6 on page 14).

2. Select a group from the **Encryption Center Devices** table, then select **Group** > **Security** from the menu task bar.

   The **Encryption Group Properties** dialog box displays with the **Security** tab selected.

3. Select **Backup Master Key** as the **Master Key Action**.

   The **Backup Master Key for Encryption Group** dialog box displays (Figure 68).



Backup Destination: A Recovery Set of Smart Cards

You will need a card reader attached to the management Station

1) Select Recovery Card Set Size: 1

2) Insert each card, in turn, into the card reader and wait for its ID to appear below.

3) Enter card assignment information. First and Last names are required.

4) Enter card password below.

Capitalization matters, 6-64 characters.

Completed	Card	Card ID	First Name	Last Name	Notes

Status Waiting for card to be inserted ...

FIGURE 68 Backup Destination (to smart cards) dialog box

4. Select **A Recovery Set of Smart Cards** as the **Backup Destination**.

5. Enter the recovery card set size.

6. Insert the first blank card and wait for the card serial number to appear.

7. Run the additional cards through the reader that are needed for the set. As you read each card, the card ID displays in the **Card Serial#** field. Be sure to wait for the ID to appear.

8. Enter the mandatory last name and first name of the person to whom the card is assigned.
9. Enter a Card **Password**.
10. Re-enter the password for verification.
11. Record and store the password in a secure location.
12. Click **Write Card**.

    You are prompted to insert the next card, up to the number of cards specified in step 5.
13. Repeat step 6 through step 12 for each card in the set.
14. After the last card is written, click **OK** in the **Master Key Backup** dialog box to finish the operation.

Saving a master key to a smart card set - Overview

A card reader must be attached to the SAN Management application PC to save a master key to a recovery card. Recovery cards can only be written once to back up a single master key. Each master key backup operation requires a new set of previously unused smart cards.

NOTE
Windows operating systems do not require smart card drivers to be installed separately; the driver is bundled with the operating system. However, you must install a smart card driver for Unix operating systems. For instructions, refer to the *Installation Guide*.

The key is divided among the cards in the card set, up to 10. The quorum of cards required to restore the master key must be less than the total number of cards in the set, and no greater than five. For example, when the master key is backed up to a set of three cards, a quorum of any two cards can be used together to restore the master key. When the master key is backed up to a set of 10 cards, a quorum size of up to five cards can be configured for restoring the master key. Backing up the master key to multiple recovery cards is the recommended and most secure option.

NOTE
When you write the key to the card set, be sure you write the full set without canceling. If you cancel, all previously written cards become unusable; you will need to discard them and create a new set.

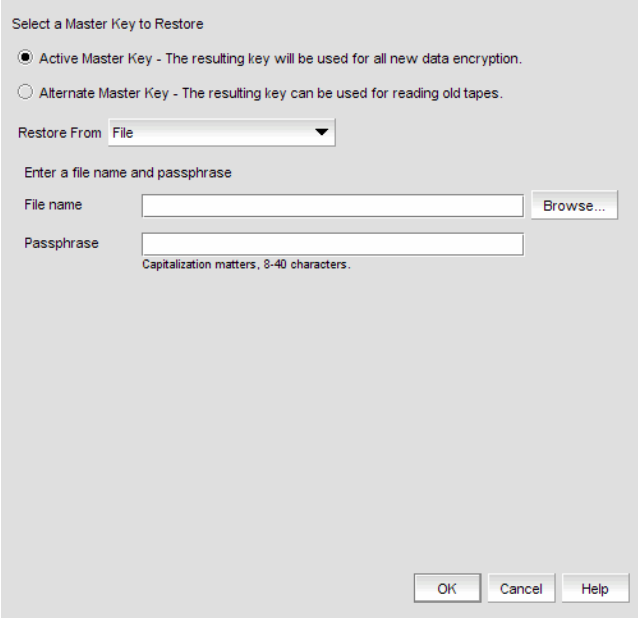
Restoring a master key from a file

Use the following procedure to restore the master key from a file.

1. Select **Configure** > **Encryption** from the menu task bar to display the **Encryption Center** dialog box (Refer to Figure 6 on page 14).
2. Select a group from the **Encryption Center Devices** table, then select **Group** > **Security** from the menu task bar.

    The **Encryption Group Properties** dialog box displays with the **Security** tab selected.
3. Select **Restore Master Key** as the **Master Key Action**.

    The **Restore Master Key for Encryption Group** dialog box displays (Figure 69).



Select a Master Key to Restore

● Active Master Key - The resulting key will be used for all new data encryption.

○ Alternate Master Key - The resulting key can be used for reading old tapes.

Restore From: File

Enter a file name and passphrase

File name:

Passphrase:

Capitalization matters, 8-40 characters.

FIGURE 69 **Select a Master Key to Restore (from file) dialog box**

4. Choose the active or alternate master key for restoration, as appropriate.
5. Select **File** as the **Restore From** location.
6. Enter a file name, or browse to the desired location.
7. Enter the passphrase. The passphrase that was used to back up the master key must be used to restore the master key.
8. Click **OK**.

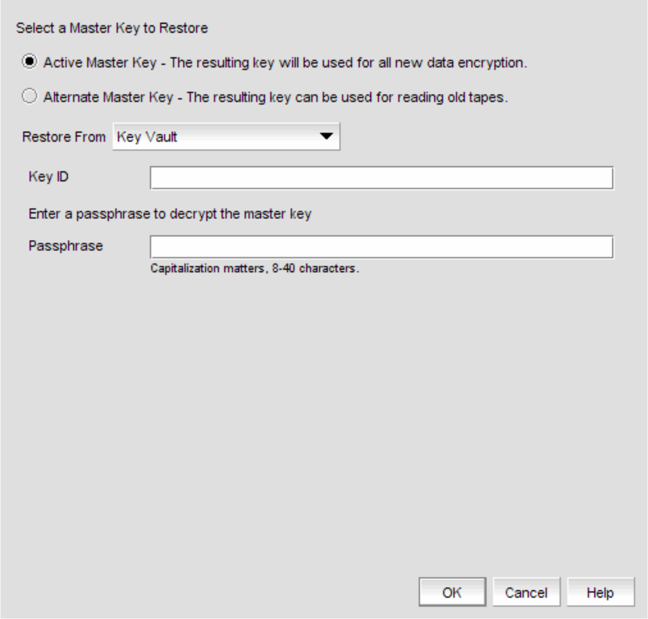
Restoring a master key from a key vault

Use the following procedure to restore the master key from a key vault:

1. Select **Configure > Encryption** from the menu task bar to display the **Encryption Center** dialog box (Refer to Figure 6 on page 14).
2. Select a group from the **Encryption Center Devices** table, then select **Group > Security** from the menu task bar.

   The **Encryption Group Properties** dialog box displays with the **Security** tab selected.
3. Select **Restore Master Key** as the **Master Key Action.**

   The **Restore Master Key for Encryption Group** dialog box displays (Figure 70).



Select a Master Key to Restore

● Active Master Key - The resulting key will be used for all new data encryption.

○ Alternate Master Key - The resulting key can be used for reading old tapes.

Restore From: Key Vault

Key ID

Enter a passphrase to decrypt the master key

Passphrase

Capitalization matters, 8-40 characters.

FIGURE 70 Select a Master Key to Restore (from key vault) dialog box

4. Choose the active or alternate master key for restoration, as appropriate.
5. Select **Key Vault** as the **Restore From** location.
6. Enter the key ID of the master key that was backed up to the key vault.
7. Enter the passphrase. The passphrase that was used to back up the master key must be used to restore the master key.
8. Click **OK**.

Restoring a master key from a smart card set

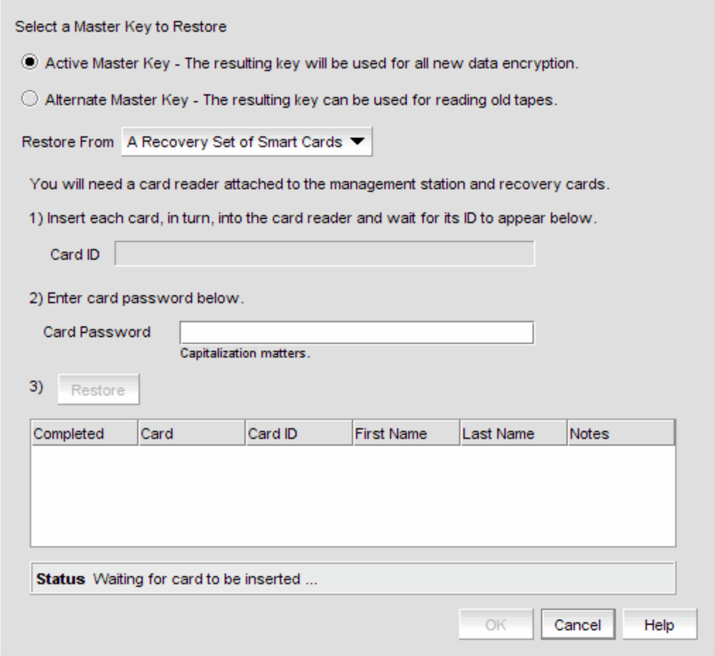
A card reader must be attached to the SAN Management application PC to complete this procedure.

Use the following procedure to restore the master key from a set of smart cards.

1. Select **Configure** > **Encryption** from the menu task bar to display the **Encryption Center** dialog box (Refer to Figure 6 on page 14).
2. Select a group from the **Encryption Center Devices** table, then select **Group** > **Security** from the menu task bar.

   The **Encryption Group Properties** dialog box displays with the **Security** tab selected.
3. Select **Restore Master Key** as the **Master Key Action**.

   The **Restore Master Key for Encryption Group** dialog box displays (Figure 71).



Select a Master Key to Restore

● Active Master Key - The resulting key will be used for all new data encryption.

○ Alternate Master Key - The resulting key can be used for reading old tapes.

Restore From: A Recovery Set of Smart Cards

You will need a card reader attached to the management station and recovery cards.

1) Insert each card, in turn, into the card reader and wait for its ID to appear below.

Card ID

2) Enter card password below.

Card Password

Capitalization matters.

3) Restore

Completed	Card	Card ID	First Name	Last Name	Notes

Status Waiting for card to be inserted ...

FIGURE 71 Select a Master Key to Restore (from a recovery set of smart cards) dialog box

4. Choose the active or alternate master key for restoration, as appropriate.
5. Select **A Recovery Set of Smart Cards** as the **Restore From** location.
6. Insert the recovery card containing a share of the master key that was backed up earlier, and wait for the card serial number to appear.
7. Enter the password that was used to create the card. After five unsuccessful attempts to enter the correct password, the card becomes locked and unusable.
8. Click **Restore**.

   You are prompted to insert the next card, if needed.
9. Repeat step 6 through step 8 until all cards in the set have been read.
10. Click **OK**.

Creating a master key

1. Select **Configure** > **Encryption** from the menu task bar to display the **Encryption Center** dialog box (Refer to Figure 6 on page 14).
2. Select a group from the **Encryption Center Devices** table, then select **Group** > **Security** from the menu task bar.

   The **Encryption Group Properties** dialog box displays with the **Security** tab selected.
3. Select **Create a New Master Key** from the list.

   A warning dialog displays.
4. Click **Yes** to proceed.

Security Settings

Security settings help you identify if system cards are required to initialize an encryption engine and also determine the number of authentication cards needed for a quorum.

1. Select **Configure > Encryption** from the menu task bar to display the **Encryption Center** dialog box (Refer to Figure 6 on page 14).
2. Select a group from the **Encryption Center Devices** table, then select **Group > Security** from the menu task bar.

   The **Select Security Settings** dialog box displays. The dialog box contains the following information:

   - **Quorum Cards**: Select the number of authentication cards needed for a quorum. The quorum is always set to one card less than the number of cards registered. For example, if you register three cards, the quorum needed for authentication is two.
   - **System Cards:** Determine whether or not a system card is required to initialize the encryption engine

NOTE
The Select Security Settings dialog box only sets a quorum number for authentication cards. To register authentication cards, click **Next** to display the **Authentication Cards** dialog box.

Zeroizing an encryption engine

Zeroizing is the process of erasing all data encryption keys and other sensitive encryption information in an encryption engine. You can zeroize an encryption engine manually to protect encryption keys. No data is lost because the data encryption keys for the encryption targets are stored in the key vault.

Zeroizing has the following effects:

- All copies of data encryption keys kept in the encryption switch or blade are erased.
- Internal public and private key pairs that identify the encryption engine are erased and the encryption switch or blade is in the FAULTY state.
- All encryption operations on this engine are stopped and all virtual initiators (VI) and virtual targets (VT) are removed from the fabric's name service.
- The key vault link key (for NetApp LKM key vaults) or the master key (for other key vaults) is erased from the encryption engine.

  Once enabled, the encryption engine is able to restore the necessary data encryption keys from the key vault when the link key (for the NetApp Lifetime Key Management application) or the master key (for other key vaults) is restored.
- If the encryption engine was part of an HA cluster, targets fail over to the peer, which assumes the encryption of all storage targets. Data flow will continue to be encrypted.
- If there is no HA backup, host traffic to the target will fail as if the target has gone offline. The host will not have unencrypted access to the target. There will be no data flow at all because the encryption virtual targets will be offline.

NOTE
Zeroizing an engine affects the I/Os, but all target and LUN configuration remain intact. Encryption target configuration data is not deleted.

You can zeroize an encryption engine only if it is enabled (running), or disabled but ready to be enabled. If the encryption engine is not in one of these states, an error message results.

When using an opaque key vault, if all encryption engines in an encryption group are zeroized, the encryption group loses the master key required to read data encryption keys from the key vault. After the encryption engines are rebooted and re-enabled, you must restore the master key from a backup copy, or alternatively, you can generate a new master key and back it up. Restoring the master key from a backup copy or generating a new master key and backing it up indicates that all previously generated DEKs will not be decryptable unless the original master key used to encrypt them is restored.

Setting zeroization

Use the **Restore Master key** wizard from the **Encryption Group Properties** dialog box to restore the master key from a backup copy.

1. Select **Configure > Encryption** from the menu task bar to display the **Encryption Center** dialog box (Refer to Figure 6 on page 14).
2. Select an encryption engine from the **Encryption Center Devices** table, then select **Engine > Zeroize** from the menu task bar.

   A warning dialog box describes consequences and actions required to recover.
3. Click **Yes** to zeroize the encryption engine.
   - For an encryption blade: After the zeroize operation is successful, a message displays noting that the encryption blade will be powered off and powered on to make it operational again. Click **OK** to close the message. After the encryption blade is powered on, click **Refresh** in the **Encryption Center** dialog box to update the status of the encryption blade and perform any operations.
   - For an encryption switch: After the zeroization operation is successful, you are instructed to reboot the encryption switch. Click **OK** to close the message, then reboot the encryption switch. After the encryption switch is rebooted, click **Refresh** in the **Encryption Center** dialog box to update the status of the encryption switch and perform any operations.

Using the Encryption Targets dialog box

The **Encryption Targets** dialog box enables you to send outbound data that you want to store as ciphertext to an encryption device. The encryption target acts as a virtual target when receiving data from a host, and as a virtual initiator when writing the encrypted data to storage.

NOTE
The **Encryption Targets** dialog box enables you to launch a variety of wizards and other related dialog boxes.

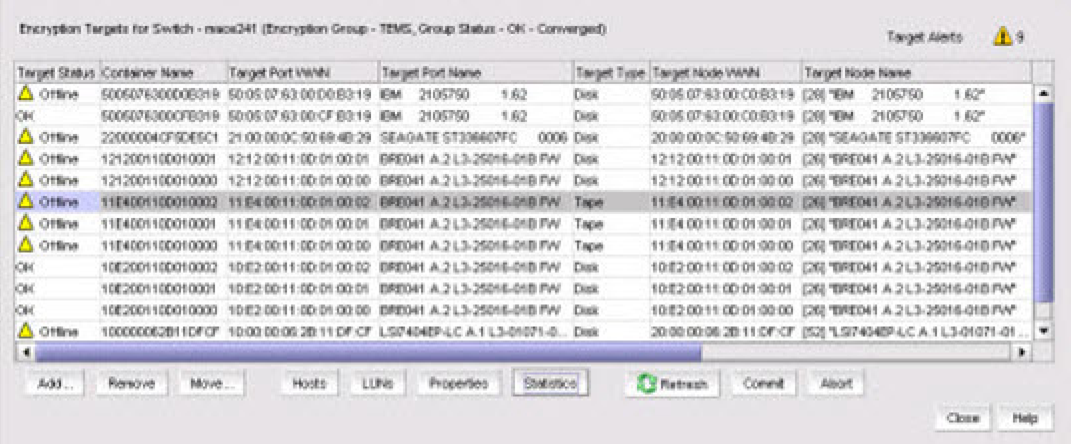
To access the Encryption Targets dialog box, complete the following steps.

1. Select **Configure** > **Encryption** from the menu task bar to display the **Encryption Center** dialog box (Refer to Figure 6 on page 14).

2. Select a group, switch, or engine from the **Encryption Center Devices** table, then select **Group/Switch/Engine** > **Targets** from the menu task bar.

   NOTE
   You can also select a group, switch, or engine from the **Encryption Center Devices** table, then click the **Targets** icon.

   The **Encryption Targets** dialog box displays (Figure 72). The targets currently being encrypted by the selected group, switch, or encryption engine are listed. If a group is selected, all configured targets in the group are displayed. If a switch is selected, all configured targets for the switch are displayed.



Encryption Targets for Switch - maca241 (Encryption Group - TEMS, Group Status - OK - Converged)

Target Status	Container Name	Target Port WWN	Target Port Name	Target Type	Target Node WWN	Target Node Name
⚠ Offline	5005076300D0B319	50:05:07:63:00:D0:B3:19	IBM 2105750 1.62	Disk	50:05:07:63:00:C0:B3:19	[28] "IBM 2105750 1.62"
OK	5005076300CFB319	50:05:07:63:00:CF:B3:19	IBM 2105750 1.62	Disk	50:05:07:63:00:C0:B3:19	[28] "IBM 2105750 1.62"
⚠ Offline	22000004CF9DE5C1	21:00:00:0C:50:69:4B:29	SEAGATE ST336607FC 0006	Disk	20:00:00:0C:50:69:4B:29	[28] "SEAGATE ST336607FC 0006"
⚠ Offline	121200110D010001	12:12:00:11:0D:01:00:01	BRE041 A.2 L3-25016-01B FW	Disk	12:12:00:11:0D:01:00:01	[26] "BRE041 A.2 L3-25016-01B FW"
⚠ Offline	121200110D010000	12:12:00:11:0D:01:00:00	BRE041 A.2 L3-25016-01B FW	Disk	12:12:00:11:0D:01:00:00	[26] "BRE041 A.2 L3-25016-01B FW"
⚠ Offline	11E400110D010002	11:E4:00:11:0D:01:00:02	BRE041 A.2 L3-25016-01B FW	Tape	11:E4:00:11:0D:01:00:02	[26] "BRE041 A.2 L3-25016-01B FW"
⚠ Offline	11E400110D010001	11:E4:00:11:0D:01:00:01	BRE041 A.2 L3-25016-01B FW	Tape	11:E4:00:11:0D:01:00:01	[26] "BRE041 A.2 L3-25016-01B FW"
⚠ Offline	11E400110D010000	11:E4:00:11:0D:01:00:00	BRE041 A.2 L3-25016-01B FW	Tape	11:E4:00:11:0D:01:00:00	[26] "BRE041 A.2 L3-25016-01B FW"
OK	10E200110D010002	10:E2:00:11:0D:01:00:02	BRE041 A.2 L3-25016-01B FW	Disk	10:E2:00:11:0D:01:00:02	[26] "BRE041 A.2 L3-25016-01B FW"
OK	10E200110D010001	10:E2:00:11:0D:01:00:01	BRE041 A.2 L3-25016-01B FW	Disk	10:E2:00:11:0D:01:00:01	[26] "BRE041 A.2 L3-25016-01B FW"
OK	10E200110D010000	10:E2:00:11:0D:01:00:00	BRE041 A.2 L3-25016-01B FW	Disk	10:E2:00:11:0D:01:00:00	[26] "BRE041 A.2 L3-25016-01B FW"
⚠ Offline	100000062B11DFCF	10:00:00:06:2B:11:DF:CF	LSI7404EP-LC A.1 L3-01071-0...	Disk	20:00:00:06:2B:11:DF:CF	[52] "LSI7404EP-LC A.1 L3-01071-01 ...

FIGURE 72 Encryption Targets dialog box

Redirection zones

It is recommended that you configure the host and target in the same zone *before* you configure them for encryption. Doing so creates a redirection zone to redirect the host/target traffic through the encryption engine; however, a redirection zone can only be created if the host and target are in the same zone. If the host and target are not already configured in the same zone, you can configure them for encryption, but you will still need to configure them in the same zone, which will then enable you to create the redirection zone as a separate step.

NOTE
If the encryption group is busy when you click **Commit**, you are given the option to either force the commit, or abort the changes. Click **Commit** to re-create the redirection zone.

Disk device decommissioning

A disk device needs to be decommissioned when any of the following occurs:

- The storage lease expires for an array, and devices must be returned or exchanged.
- Storage is reprovisioned for movement between departments.
- An array or device is removed from service.

In all cases, all data on the disk media must be rendered inaccessible. Device decommissioning deletes all information that could be used to recover the data, for example, information related to master key IDs and cache files.

After device decommissioning is performed, the following actions occur:

- Metadata on the LUN is erased and the reference is removed from cache on the Brocade Encryption Switch.
- The LUN state is shown as decommissioned in the key vault.
- The LUN is removed from the container.

NOTE
The key IDs that were used for encrypting the data are returned.

When disk LUNs are decommissioned, the decommissioned keys are still stored on the switch. In order to delete them from the switch, you must view them from the **Decommissioned Key IDs** dialog box. Refer to Figure 73.

When a device decommission operation fails on the encryption group leader for any reason, the crypto configuration remains uncommitted until a user-initiated commit or a subsequent device decommission operation issued on the encryption group leader completes successfully. Device decommission operations should always be issued from a committed configuration. If not, the operation will fail with the error message **An outstanding transaction is pending in Switch/EG**. If this occurs, you can resolve the problems by committing the configuration from the encryption group leader.

Provided that the crypto configuration is not left uncommitted because of any crypto configuration changes or a failed device decommission operation issued on a encryption group leader node, this error message will not be seen for any device decommission operation issued serially on an encryption group member node. If more than one device decommission operation is attempted in an encryption group from member nodes simultaneously, this error message is transient and will go away after device decommission operation is complete. If the device decommissioning operation fails, retry the operation after some time has passed.

Decommissioning disk LUNs

Use the following procedure to decommission a disk LUN.

1. Select **Configure > Encryption** from the menu task bar to display the **Encryption Center** dialog box (Refer to Figure 6 on page 14).
2. Select a group, switch, or engine from the **Encryption Center Devices** table that contains the storage device to be configured, then select **Group/Switch/Engine > Targets** from the menu task bar.

   NOTE
   You can also select a group, switch, or engine from the **Encryption Center Devices** table, then click the **Targets** icon.

   The **Encryption Targets** dialog box displays (Figure 60).
3. Select a Target storage device from the list, then click **LUNs**.

   The **Encryption Target Disk LUNs** dialog box displays.
4. Select the LUNs associated with the device, then click **Decommission**.

   A warning message displays.
5. Click **Yes** to proceed with the decommissioning process.

   A **LUN Decommission Status** dialog box is displayed while the LUNs are being decommissioned. Click **OK** to close the dialog box.

   If a rekey operation is currently in progress on a selected LUN, a message is displayed that gives you a choice of doing a **Forced Decommission**, or to **Cancel** and try later after the rekey operation is complete.
6. To check on the progress of the decommissioning operation, click **Refresh**. When decommissioning is complete, the LUNs are removed from the **Encryption Target LUNs** table.

Displaying and deleting decommissioned key IDs

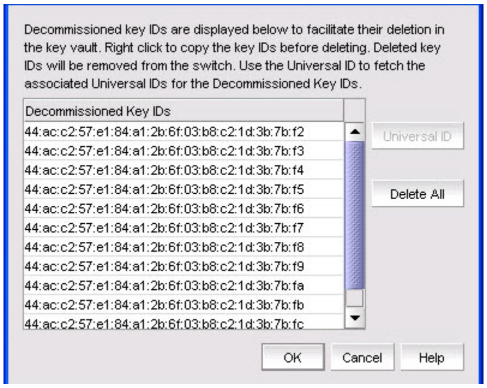
When disk LUNs are decommissioned, the process includes the disabling of the key record in the key vault and indication that the key has been decommissioned. These decommissioned keys are still stored on the switch. You can display, copy, and delete them as an additional security measure.

The **Decommissioned Key IDs** dialog box lists Key IDs that have been decommissioned at the key vault. They should also be deleted from the switch for added security, and to create room for new key IDs. Using this dialog box, you can delete key IDs that are decommissioned at the key vault, but still stored on the switch.

In order to delete keys from the key vault, you need to know the Universal ID (UUID). To display vendor-specific UUIDs of decommissioned key IDs, complete the following procedure:

1. Select **Configure** > **Encryption** from the menu task bar to display the **Encryption Center** dialog box (Refer to Figure 6 on page 14).
2. Select a switch from the **Encryption Center Devices table**, then select **Switch** > **Decommissioned key IDs** from the menu task bar.

   The **Decommissioned Key IDs** dialog box displays (Figure 73).



Decommissioned key IDs are displayed below to facilitate their deletion in the key vault. Right click to copy the key IDs before deleting. Deleted key IDs will be removed from the switch. Use the Universal ID to fetch the associated Universal IDs for the Decommissioned Key IDs.

Decommissioned Key IDs
44:ac:c2:57:e1:84:a1:2b:6f:03:b8:c2:1d:3b:7b:f2
44:ac:c2:57:e1:84:a1:2b:6f:03:b8:c2:1d:3b:7b:f3
44:ac:c2:57:e1:84:a1:2b:6f:03:b8:c2:1d:3b:7b:f4
44:ac:c2:57:e1:84:a1:2b:6f:03:b8:c2:1d:3b:7b:f5
44:ac:c2:57:e1:84:a1:2b:6f:03:b8:c2:1d:3b:7b:f6
44:ac:c2:57:e1:84:a1:2b:6f:03:b8:c2:1d:3b:7b:f7
44:ac:c2:57:e1:84:a1:2b:6f:03:b8:c2:1d:3b:7b:f8
44:ac:c2:57:e1:84:a1:2b:6f:03:b8:c2:1d:3b:7b:f9
44:ac:c2:57:e1:84:a1:2b:6f:03:b8:c2:1d:3b:7b:fa
44:ac:c2:57:e1:84:a1:2b:6f:03:b8:c2:1d:3b:7b:fb
44:ac:c2:57:e1:84:a1:2b:6f:03:b8:c2:1d:3b:7b:fc

FIGURE 73 Decommissioned Key IDs dialog box

The dialog box contains the following information:

- **Decommissioned key IDs** that have been decommissioned at the key vault are listed in a table.
- **Universal ID** button: Launches the **Universal ID** dialog box to display the universal ID for each selected decommissioned key.

  You need to know the Universal ID (UUID) associated with the decommissioned disk LUN key IDs in order to delete keys from the key vault. You can display vendor-specific UUIDs of decommissioned key IDs. For more information, refer to "Displaying Universal IDs" on page 100.
- **Delete All** button: Deletes all of the listed decommissioned key IDs.

3. Click **Delete All** to delete the decommissioned keys from the switch. As a precaution, copy the keys to a secure location before deleting them from the switch. Right-click on an entry in the table to individually select a key ID. You may also copy or export a single row within the table or the entire table. To export the keys, right-click and select **Export,** which will export the key IDs.

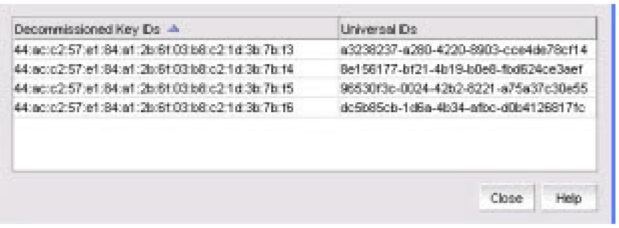
Displaying Universal IDs

In order to delete keys from the key vaults, you need to know the Universal ID (UUID) associated with the decommissioned disk LUN key IDs. To display the Universal IDs, complete the following procedure:

1. Select **Configure** > **Encryption** from the menu task bar to display the **Encryption Center** dialog box (Refer to Figure 6 on page 14).
2. Select a switch from the **Encryption Center Devices table**, then select **Switch** > **Decommissioned key IDs** from the menu task bar.

   The **Decommissioned Key IDs** dialog box displays (Refer to Figure 73).
3. Select the desired decommissioned key IDs from the **Decommissioned Key IDs** table, then click **Universal ID**.

   The **Universal IDs** dialog box displays the universal ID for each selected decommissioned key (Figure 74).



Decommissioned Key IDs ▲	Universal IDs
44:ac:c2:57:e1:84:a1:2b:61:03:b8:c2:1d:3b:7b:13	a3238237-a280-4220-8903-cce4de78cf14
44:ac:c2:57:e1:84:a1:2b:61:03:b8:c2:1d:3b:7b:14	8e156177-bf21-4b19-b0e8-fbd624ce3aef
44:ac:c2:57:e1:84:a1:2b:61:03:b8:c2:1d:3b:7b:15	96530f3c-0024-42b2-8221-a75a37c30e55
44:ac:c2:57:e1:84:a1:2b:61:03:b8:c2:1d:3b:7b:16	dc5b85cb-1d6a-4b34-afbc-d0b41268171c

FIGURE 74 Universal IDs dialog box

4. Click **Close**.

NOTE
You will need to export the decommissioned key ID to the key vault.

Rekeying all disk LUNs manually

The encryption management application allows you to perform a manual rekey operation on all encrypted primary disk LUNs and all non-replicated disk LUNs hosted on the encryption node that are in the read-write state.

Manual rekeying of all LUNs might take an extended period of time. BNA allows manual rekey of no more than 10 LUNs concurrently. If the node has more than 10 LUNs, additional LUN rekey operations will remain in the pending state until others have finished.

The following conditions must be satisfied for the manual rekey operation to run successfully:

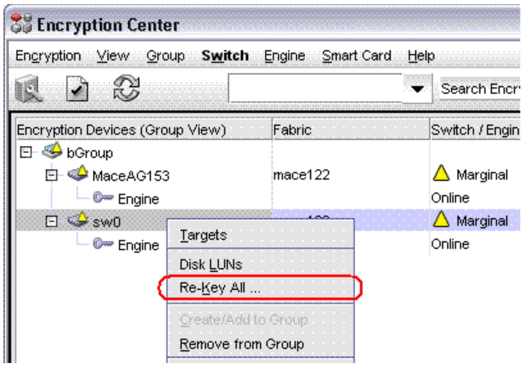
- The node on which you perform the manual rekey operation must be a member of an encryption group, and that encryption group must have a key vault configured.
- The node must be running Fabric OS 7.0.0 or later.
- The encryption group must be in the converged state.
- The target container that hosts the LUN must be online.

In addition to providing the ability to launch manual rekey operations, BNA also enables you to monitor their progress.

Setting disk LUN Re-key All

To rekey all disk LUNs on an encryption node, complete these steps:

1. Select **Configure** > **Encryption** from the menu task bar to display the **Encryption Center** dialog box (Refer to Figure 6 on page 14).
2. Select the switch on which to perform a manual re-key from the **Encryption Center Devices** table, then select **Switch** > **Re-Key All** from the menu task bar (Figure 75).



Encryption Devices (Group View)	Fabric	Switch / Engin
bGroup		
MaceAG153	mace122	Marginal
Engine		Online
sw0		Marginal
Engine		Online

FIGURE 75 Selecting the Re-Key All operation

If REPL support is enabled on the encryption group, a confirmation dialog box displays, asking whether to rekey mirror LUNs.

3. Click **Yes** to includes mirror LUNs, or click **No** to exclude mirror LUNs.

   A warning message displays, requesting confirmation to proceed with the rekey operation.

4. Click **Yes.**

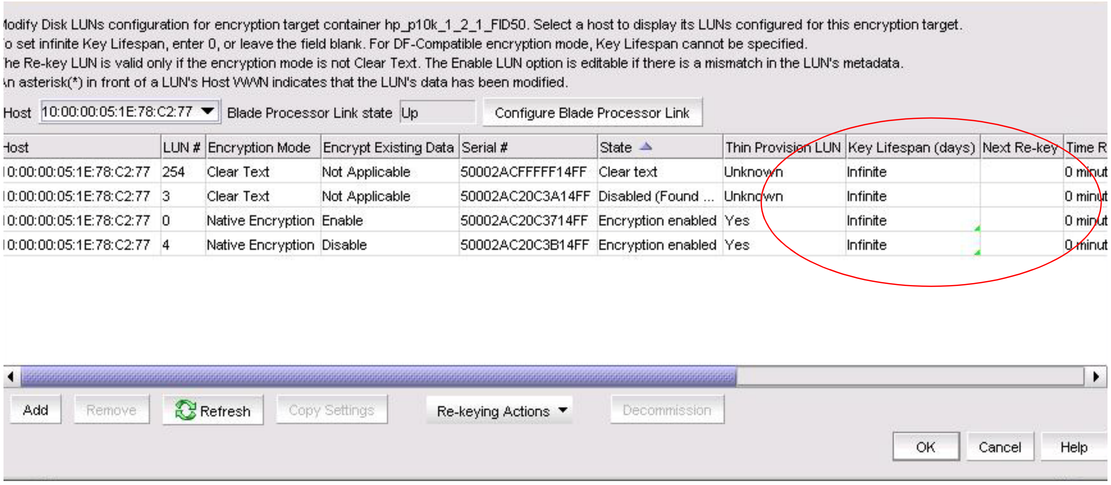
   Rekeying operations begin on up to 10 LUNs. If more than 10 LUNs are configured on the switch, the remaining rekey operations are held in the pending state.

5. Open the **Encryption Target Disk LUNs** dialog box to see LUNs being rekeyed and LUNs pending.
   a. Select **Configure** > **Encryption** from the menu task bar to display the **Encryption Center** dialog box (Refer to Figure 6 on page 14).
   b. Select the encryption switch from the **Encryption Center Devices** table, then select **Targets** from the menu task bar.

      The **Encryption Targets** dialog box displays (Refer to Figure 47 on page 64).

6. Select a disk LUN device from the table, then click **LUNs.**

   The **Encryption Targets Disk LUNs** dialog box displays (Figure 76).The dialog box lists the status of the rekey operation.



lodify Disk LUNs configuration for encryption target container hp_p10k_1_2_1_FID50. Select a host to display its LUNs configured for this encryption target.
'o set infinite Key Lifespan, enter 0, or leave the field blank. For DF-Compatible encryption mode, Key Lifespan cannot be specified.
'he Re-key LUN is valid only if the encryption mode is not Clear Text. The Enable LUN option is editable if there is a mismatch in the LUN's metadata.
\n asterisk(*) in front of a LUN's Host WWN indicates that the LUN's data has been modified.

Host 10:00:00:05:1E:78:C2:77 Blade Processor Link state Up

Host	LUN #	Encryption Mode	Encrypt Existing Data	Serial #	State	Thin Provision LUN	Key Lifespan (days)	Next Re-key	Time R
10:00:00:05:1E:78:C2:77	254	Clear Text	Not Applicable	50002ACFFFFF14FF	Clear text	Unknown	Infinite		0 minut
10:00:00:05:1E:78:C2:77	3	Clear Text	Not Applicable	50002AC20C3A14FF	Disabled (Found ...	Unknown	Infinite		0 minut
10:00:00:05:1E:78:C2:77	0	Native Encryption	Enable	50002AC20C3714FF	Encryption enabled	Yes	Infinite		0 minut
10:00:00:05:1E:78:C2:77	4	Native Encryption	Disable	50002AC20C3B14FF	Encryption enabled	Yes	Infinite		0 minut

FIGURE 76 Pending manual rekey operations

Viewing disk LUN rekeying details

You can view details related to the rekeying of a selected target disk LUN from the **LUN Re-keying Details** dialog box.

1. Select **Configure** > **Encryption** from the menu task bar to display the **Encryption Center** dialog box (Refer to Figure 6 on page 14).

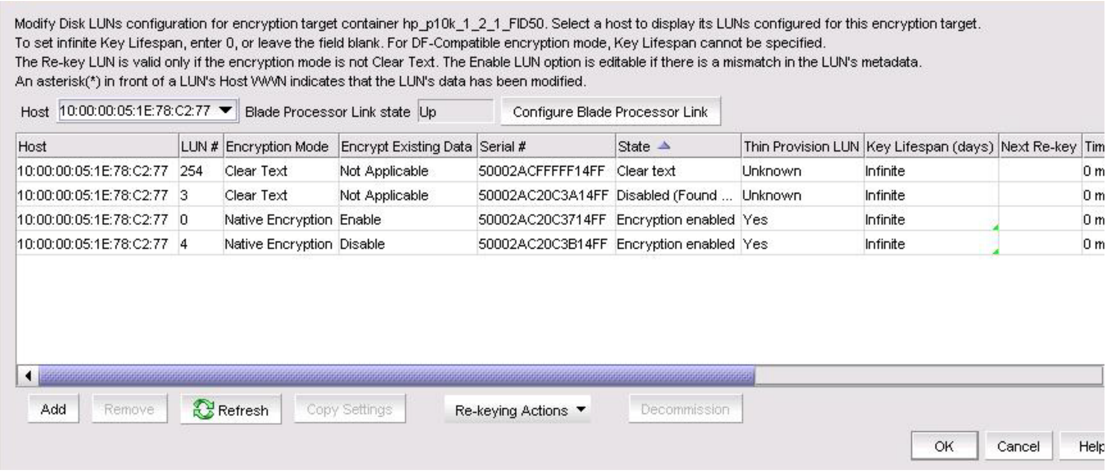
2. Select a group, switch, or engine from the **Encryption Center Devices** table, then select **Group/Switch/Engine** > **Targets**, or right-click the group, switch, or engine and select **Targets**.

   NOTE
   You can also select a group, switch, or engine from the **Encryption Center Devices** table, then click the **Targets** icon.

   The **Encryption Targets** dialog box displays.

3. Select a Target storage device, then **select Group/Switch/Engine** > **Disk LUNs.**

   The **Encryption Target Disk LUNs** dialog box displays (Figure 77). Initially the list is empty. You must add LUNs manually.



Modify Disk LUNs configuration for encryption target container hp_p10k_1_2_1_FID50. Select a host to display its LUNs configured for this encryption target.
To set infinite Key Lifespan, enter 0, or leave the field blank. For DF-Compatible encryption mode, Key Lifespan cannot be specified.
The Re-key LUN is valid only if the encryption mode is not Clear Text. The Enable LUN option is editable if there is a mismatch in the LUN's metadata.
An asterisk(*) in front of a LUN's Host WWN indicates that the LUN's data has been modified.

Host: 10:00:00:05:1E:78:C2:77 Blade Processor Link state: Up

Host	LUN #	Encryption Mode	Encrypt Existing Data	Serial #	State	Thin Provision LUN	Key Lifespan (days)	Next Re-key	Tim
10:00:00:05:1E:78:C2:77	254	Clear Text	Not Applicable	50002ACFFFFF14FF	Clear text	Unknown	Infinite		0 m
10:00:00:05:1E:78:C2:77	3	Clear Text	Not Applicable	50002AC20C3A14FF	Disabled (Found ...	Unknown	Infinite		0 m
10:00:00:05:1E:78:C2:77	0	Native Encryption	Enable	50002AC20C3714FF	Encryption enabled	Yes	Infinite		0 m
10:00:00:05:1E:78:C2:77	4	Native Encryption	Disable	50002AC20C3B14FF	Encryption enabled	Yes	Infinite		0 m

FIGURE 77 Encryption Target Disk LUNs dialog box

4. Click **Add**.

   The **Add Disk LUNs** dialog box displays. This dialog box includes a table of all LUNs in the storage device that are visible to the hosts.

5. Click **Re-keying Details**.

   The **LUN Re-keying Details** dialog box displays. The dialog box contains the following information:

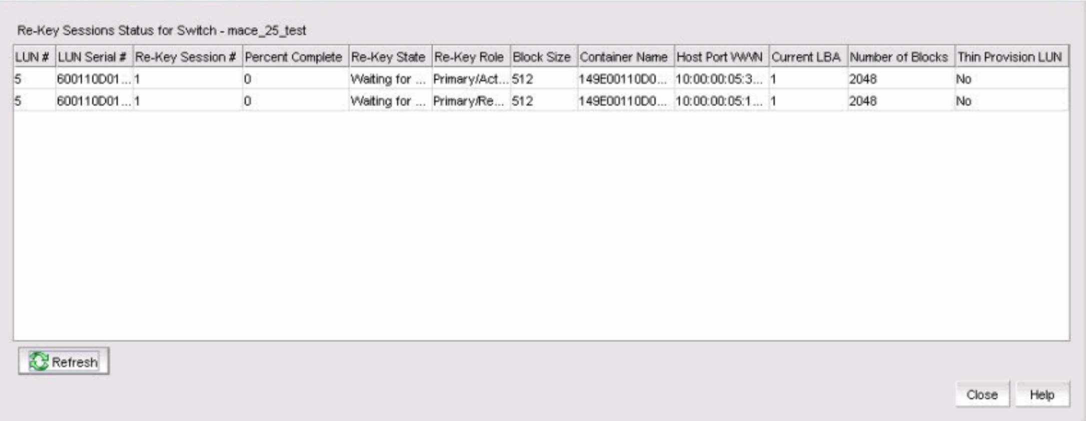
   - **Key ID**: The LUN key identifier.
   - **Key ID State**: The state of the LUN rekeying operation.
   - **Encryption Algorithm**: The algorithm of the LUN rekeying operation.
   - **Re-key Session Number**: The session number of the LUN rekeying operation.
   - **Re-key Role**: The role of the LUN rekeying operation.
   - **Re-key State**: The state of a manual LUN rekeying operation. Options are:
     - **Read Phase**
     - **Write Phase**
     - **Pending**
     - **Disabled**
   - **Block Size**: The block size used on the LUN.
   - **Number of Blocks**: The number of blocks written.
   - **Current LBA**: The Logical Block Address (LBA) of the block that is currently being written.
   - **Re-key Completion**: The status of the LUN rekeying operation's progress.

Viewing the progress of manual rekey operations

To monitor the progress of manual rekey operations, complete these steps:

1. Select **Configure** > **Encryption** from the menu task bar to display t**he Encryption Center** dialog box (Refer to Figure 6 on page 14).
1. Select an encryption group from the **Encryption Center Devices** table, then select **Group** > **Re-Key Sessions** from the menu task bar.

   The **Re-Key Sessions Status** dialog box displays, which enables you to check on the status of each LUN that is being rekeyed within an encryption group (Figure 78).



Re-Key Sessions Status for Switch - mace_25_test

LUN #	LUN Serial #	Re-Key Session #	Percent Complete	Re-Key State	Re-Key Role	Block Size	Container Name	Host Port WWN	Current LBA	Number of Blocks	Thin Provision LUN
5	600110D01...	1	0	Waiting for ...	Primary/Act...	512	149E00110D0...	10:00:00:05:3...	1	2048	No
5	600110D01...	1	0	Waiting for ...	Primary/Re...	512	149E00110D0...	10:00:00:05:1...	1	2048	No

FIGURE 78 Re-Key Sessions Status dialog box

The dialog box contains the following information:

- **LUN #**: The LUN number.
- **LUN Serial #**: The LUN serial number.
- **Re-Key Session #**: The number assigned to the rekeying session.
- **Percent Complete**: The percentage of completion of the rekeying session.
- **Re-Key State**: Options are:
  - Re-Key Setup
  - LUN Prep
  - LUN Clean-up
  - Key Update
  - Read Phase
  - Write Phase
  - HA Sync Phase
- **Re-Key Role**: Options are:
  - Primary/Active
  - Backup/Active
- **Block Size**: The block size used on the LUN.
- **Container Name**: The CryptoTarget container name.
- **Host Port WWN**: The WWN of the host port that is being used for the write operation.

- **Current LBA**: The Logical Block Address (LBA) of the block that is currently being written.
- **Number of Blocks**: The number of blocks written.
- **Thin Provision LUN**: Identifies if the new LUN is a thin provisioned LUN. Options are:
  - **Yes:** Thin provision support is limited to Brocade-tested storage arrays. The thin provision LUN status will be displayed as **Yes** for supported storage arrays only.
  - **No:** Shown as No if the LUN is not a thin provisioned LUN.
  - **Unknown:** Shown if the LUN status cannot be determined.
  - **Not Applicable:** Applies to Brocade Encryption Switches that are running a Fabric OS version earlier than v7.1.0.

2. Click **Refresh** periodically to update the display.

Thin provisioned LUNs

With the introduction of Fabric OS 7.1.0, the Brocade Encryption Switch can discover if a disk LUN is a thin provisioned LUN. Support for a thin provisioned LUN is limited to disk containers only. Thin provisioned LUNs can be created with the new LUN option.

NOTE
Currently, thin provisioned LUN support is limited to Brocade-tested storage arrays running specific supported firmware releases. The thin provisioned LUN status will be displayed as **Yes** for supported storage arrays only. Contact your service representative to determine if your storage array is supported.

Thin provisioned LUNs rely on on-demand allocation of blocks of data, instead of the traditional method of allocating all blocks up front. If a thin provisioned LUN status is shown as **Yes**, then first-time encryption and rekey are done on the allocated blocks only, which results in the provisioned region of the LUN to remain the same after the rekey is performed.

Thin provisioned LUN support requires no action by the user. The Brocade Encryption Switch can automatically detect if a LUN is a thin provisioned LUN.

NOTE:

- If a LUN is a thin provisioned LUN, LUN status is shown as **Yes**. (Thin provision support is limited to Brocade-tested storage arrays. The thin provisioned LUN status will be displayed as **Yes** for supported storage arrays only.)
- If a LUN is not a thin provisioned LUN or if thin provisioning is not supported with the LUN, LUN status is shown as **No**. (This can be a result of the array not supporting thin provisioning, or the Brocade Encryption Switch/blade does not support the thin provisioning features of the array. Refer to the Fabric OS release notes for supported arrays.)
- If LUN status cannot be determined, LUN status is shown as **Unknown**.
- If you are running a Fabric OS version earlier than v7.1.0, LUN status is shown as **Not Applicable**.
- Zero detect with encryption is not supported.

Thin Provisioning support

Thin-provisioned logical unit numbers (LUNs) are increasingly used to support a pay-as-you-grow strategy for data storage capacity. Also known as dynamic provisioning, virtual LUNs, or thin LUNs, the same technology that allows storage administrators to allocate physical disk space to LUNs on an as-needed basis creates limitations around certain data-at-rest encryption operations that use the Brocade Encryption Switch or blade. Performing first-time encryption (FTE) (conversion of cleartext to ciphertext) and data rekeying operations (applying new data encryption keys to ciphertext data) on thin-provisioned LUNs results in an attempt by the encryption switch to overwrite data up to the size of the logical size of the thin-provisioned LUN, rather than limiting FTE/rekeying to the size of the physically allocated LUN size or to the data that has been written. This generally triggers the allocation of additional blocks to the thin-provisioned LUN, using up the amount of physical disk space that is available to the LUN and defeating the objective of using thin provisioning.

Additionally, for thin-provision capable storage products that support space reclamation based on data pattern recognition (for example, ‘string of zeros’), the encryption of such patterns will interfere with the space reclamation functionality of the storage and should be avoided.

Certain types of storage, including 3PAR, have been successfully tested by limiting the use of thin provisioning to “greenfield” LUNs, or LUNs that do not have any written data yet. Rekeying operations on these LUNs, like FTE, are also not permitted. As these limitations are not feasible for most environments, the recommendation from Brocade is that any encrypted LUNs be fully provisioned with disk.

Viewing time left for auto rekey

You can view the time remaining until auto rekey is no longer active for a disk LUN. The information is expressed as the difference between the next rekey date and the current date and time, and is measured in days, hours, and minutes.

Although you cannot make changes directly to the table, you can modify the time left using CLI. For more information, see the administrator’s guide supporting your key vault management system.

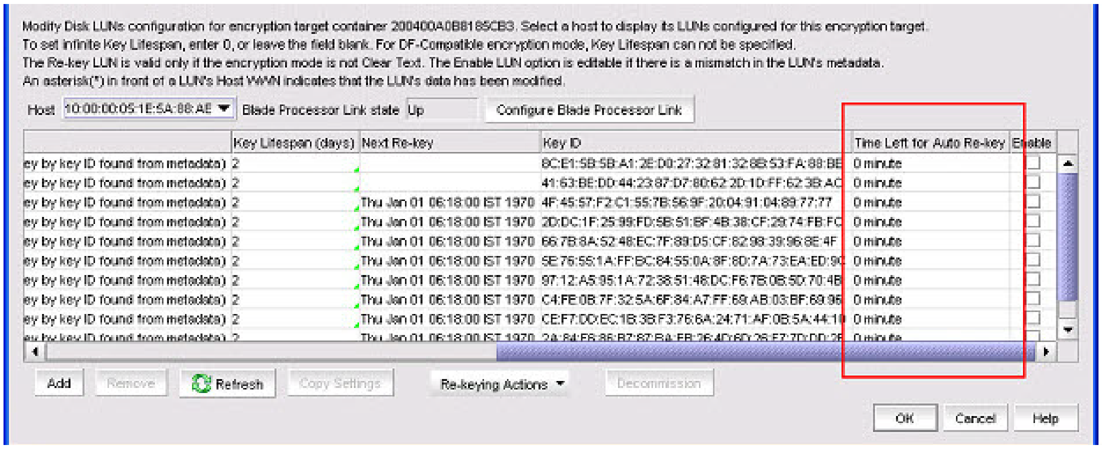
To view the time left for auto rekey, follow these steps:

1. Select **Configure > Encryption** from the menu task bar to display the **Encryption Center** dialog box. (Refer to Figure 6 on page 14.)
2. Select a group, switch, or engine from the **Encryption Center Devices** table for which to view the auto rekey information, then select **Group/Switch/Engine > Targets** from the menu task bar.

   NOTE
   You can also select a group, switch, or engine from the **Encryption Center Devices** table, then click the **Targets** icon.

   The **Encryption Targets** dialog box displays. (Refer to Figure 47.)
3. Select a target disk device from the table, then click **LUNs**.

   The **Encryption Target Disk LUNs** dialog box displays. The time left for auto rekey information is listed in the table (Figure 79).



Modify Disk LUNs configuration for encryption target container 200400A0B8185CB3. Select a host to display its LUNs configured for this encryption target.
To set infinite Key Lifespan, enter 0, or leave the field blank. For DF-Compatible encryption mode, Key Lifespan can not be specified.
The Re-key LUN is valid only if the encryption mode is not Clear Text. The Enable LUN option is editable if there is a mismatch in the LUN's metadata.
An asterisk(*) in front of a LUN's Host WWN indicates that the LUN's data has been modified.

Host 10:00:00:05:1E:5A:88:AE Blade Processor Link state Up

	Key Lifespan (days)	Next Re-key	Key ID	Time Left for Auto Re-key	Enable
ey by key ID found from metadata)	2		8C:E1:5B:5B:A1:2E:D0:27:32:81:32:8B:53:FA:88:BE	0 minute	□
ey by key ID found from metadata)	2		41:63:BE:DD:44:23:87:D7:80:62:2D:1D:FF:62:3B:AC	0 minute	□
ey by key ID found from metadata)	2	Thu Jan 01 06:18:00 IST 1970	4F:45:57:F2:C1:55:7B:56:9F:20:04:91:04:89:77:77	0 minute	□
ey by key ID found from metadata)	2	Thu Jan 01 06:18:00 IST 1970	2D:DC:1F:25:99:FD:5B:51:BF:4B:38:CF:29:74:FB:FC	0 minute	□
ey by key ID found from metadata)	2	Thu Jan 01 06:18:00 IST 1970	66:7B:8A:52:48:EC:7F:89:D5:CF:82:98:39:96:8E:4F	0 minute	□
ey by key ID found from metadata)	2	Thu Jan 01 06:18:00 IST 1970	5E:76:55:1A:FF:BC:84:55:0A:8F:8D:7A:73:EA:ED:9C	0 minute	□
ey by key ID found from metadata)	2	Thu Jan 01 06:18:00 IST 1970	97:12:A5:95:1A:72:38:51:48:DC:F6:7B:0B:5D:70:4B	0 minute	□
ey by key ID found from metadata)	2	Thu Jan 01 06:18:00 IST 1970	C4:FE:0B:7F:32:5A:6F:84:A7:FF:69:AB:03:BF:69:96	0 minute	□
ey by key ID found from metadata)	2	Thu Jan 01 06:18:00 IST 1970	CE:F7:DD:EC:1B:3B:F3:76:6A:24:71:AF:0B:5A:44:10	0 minute	□
ey by key ID found from metadata)	2	Thu Jan 01 06:18:00 IST 1970	2A:84:F6:86:B7:87:BA:FB:26:4D:6D:26:F7:7D:DD:2F	0 minute	□

FIGURE 79 Time left for auto rekey

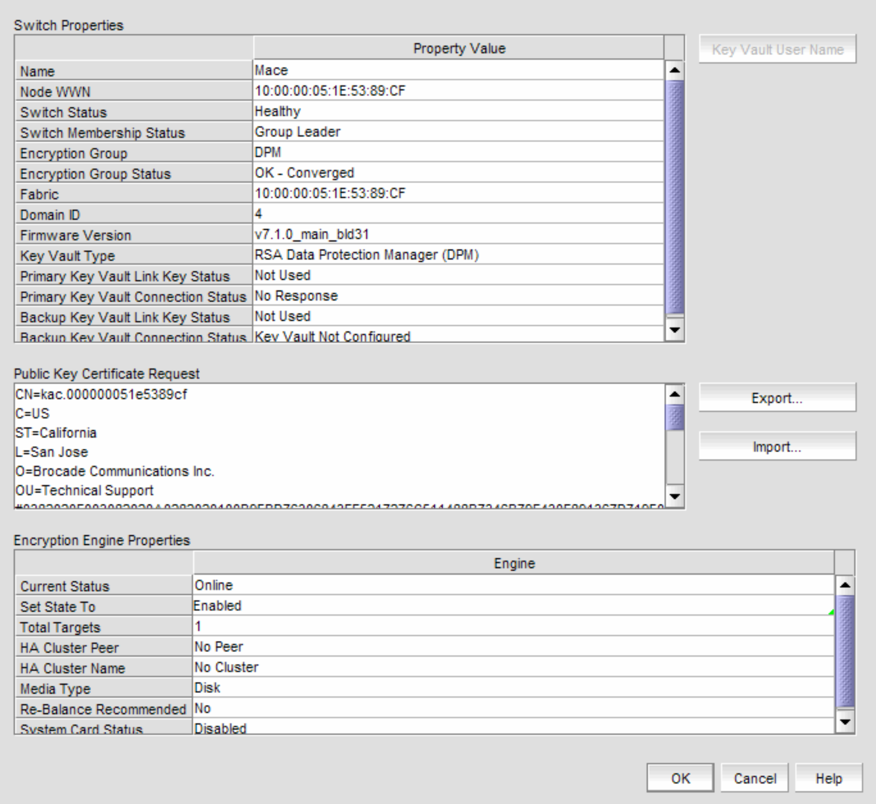
Viewing and editing switch encryption properties

To view switch encryption properties, complete the following steps:

1. Select **Configure** > **Encryption** from the menu task bar to display the **Encryption Center** dialog box. (Refer to Figure 6 on page 14.)
2. Select a switch or encryption engine from the **Encryption Center Devices** table, then select **Switch/Engine** > **Properties** from the menu task bar, or right-click a switch or encryption engine and select **Properties**.

NOTE
You can also select a group, switch, or engine from the **Encryption Center Devices** table, then click the **Properties** icon.

The **Encryption Switch Properties** dialog box displays (Figure 80).



Switch Properties

	Property Value
Name	Mace
Node WWN	10:00:00:05:1E:53:89:CF
Switch Status	Healthy
Switch Membership Status	Group Leader
Encryption Group	DPM
Encryption Group Status	OK - Converged
Fabric	10:00:00:05:1E:53:89:CF
Domain ID	4
Firmware Version	v7.1.0_main_bld31
Key Vault Type	RSA Data Protection Manager (DPM)
Primary Key Vault Link Key Status	Not Used
Primary Key Vault Connection Status	No Response
Backup Key Vault Link Key Status	Not Used
Backup Key Vault Connection Status	Key Vault Not Configured

Public Key Certificate Request

```
CN=kac.000000051e5389cf
C=US
ST=California
L=San Jose
O=Brocade Communications Inc.
OU=Technical Support
```

Encryption Engine Properties

	Engine
Current Status	Online
Set State To	Enabled
Total Targets	1
HA Cluster Peer	No Peer
HA Cluster Name	No Cluster
Media Type	Disk
Re-Balance Recommended	No
System Card Status	Disabled

FIGURE 80 Encryption Switch Properties dialog box

The dialog box contains the following information:

- **Switch Properties** table: A list of properties associated with the selected switch
  - **Name**: The name of the selected switch
  - **Node WWN**: The world wide name of the node

- Switch Status: The health status of the switch. Options are:
  - Healthy
  - Marginal
  - Down
  - Unknown
  - Unmonitored
  - Unreachable
- Switch Membership Status: The alert or informational message description, which details the health status of the switch. Options are:
  - Group Member
  - Leader-Member Comm
  - Error
  - Discovering
  - Not a member
- Encryption Group: The name of the encryption group to which the switch belongs.
- Encryption Group Status: Status options are:
  - OK/Converged: the group leader can communicate with all members
  - Degraded: the group leader cannot communicate with one or more members. The following operations are not allowed: key vault changes, master key operations, enable/disable encryption engines, Failback mode changes, HA Cluster creation or addition (removal is allowed), tape pool changes, and any configuration changes for storage targets, hosts, and LUNs.
  - Unknown: The group leader is in an unmanaged fabric.
- Fabric: The name of the fabric to which the switch belongs.
- Domain ID: The domain ID of the selected switch.
- Firmware Version: The current encryption firmware on the switch.
- Key Vault Type: Options are:
  - RSA Data Protection Manager (DPM): **NOTE:** If an encryption group contains mixed firmware nodes, the Encryption Group Properties Key Vault Type name is based on the firmware version of the group leader. For example, If a switch is running Fabric OS 7.1.0 or later, the Key Vault Type is displayed as "RSA Data Protection Manager (DPM)."If a switch is running Fabric OS prior to v7.1.0, Key Vault Type is displayed as "RSA Key Manager (RKM)".
- Primary Key Vault Link Key Status/Backup Key Vault Link Key Status:: *(LKM/SSKM key vault only.)* Shown as Not Used.
- Primary Key Vault Connection Status/Backup Key Vault Connection Status: Whether the primary key vault link is connected. Options are:
  - Unknown/Busy
  - Key Vault Not Configured
  - No Response
  - Failed authentication
  - Connected.

- Key Vault User Name button: *(TEKA key vault only.)* Shown as inactive.
- Public Key Certificate Request text box: The switch's KAC certificate signing request, which must be signed by a certificate authority (CA). The signed certificate must then be imported onto the switch and onto the primary and backup key vaults.
- Export button: Exports the public key certificate in CSR format to an external file for signing by a certificate authority (CA).
- Import button: Imports a signed public key certificate.
- Encryption Engine Properties table: The properties for the encryption engine. There may be 0 to 4 slots, one for each encryption engine in the switch.
- Current Status: The status of the encryption engine. Many possible values exist. Common options are:
  - Not Available (the engine is not initialized)
  - Disabled
  - Operational
  - need master/link key
  - Online
- Set State To: Identifies if the state is enabled or disabled. You can click the line item in the table to change the value, then click OK to apply the change.
- Total Targets: The number of encrypted target devices.
- HA Cluster Peer: The name and location of the high-availability (HA) cluster peer (another encryption engine in the same group), if in an HA configuration. If no peer is configured, No Peer is displayed.
- HA Cluster Name: The name of the HA cluster (for example, Cluster1), if in an HA configuration. HA Cluster names can have up to 31 characters. Letters, digits, and underscores are allowed.
- Media Type: The media type of the encryption engine. Options are Disk and Tape, or Disk/Tape when both are present.
- Re-Balance Recommended: Indicates if LUN rebalancing is recommended for an encryption engine that is hosting both disk and tape LUNs. Options are Yes and No.
- System Card Status: The current status of system card information for the encryption engine. Options are Enabled and Disabled.

Exporting the public key certificate signing request (CSR) from properties

To export the CSR under Public Key Certificate Request, complete the following steps.

1. Click Export, then browse to the location where you want to save the certificate and click Save.

   Alternatively, you may also copy the CSR and paste it to a file.

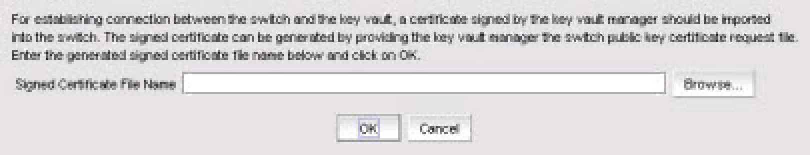
2. Submit the CSR to a certificate authority (CA) for signing. CA signing requirements and procedures differ per key manager appliance.

Importing a signed public key certificate from properties

To import a signed public key certificate, complete the following steps.

1. Click **Import**.

   The **Import Signed Certificate** dialog box displays (Figure 81).



For establishing connection between the switch and the key vault, a certificate signed by the key vault manager should be imported into the switch. The signed certificate can be generated by providing the key vault manager the switch public key certificate request file. Enter the generated signed certificate file name below and click on OK.

Signed Certificate File Name

FIGURE 81 Import Signed Certificate dialog box

2. Enter or browse to the file containing the signed certificate, then click **OK**.

   The file is imported onto the switch.

Enabling and disabling the encryption engine state from properties

To enable the encryption engine, complete the following steps:

1. Select **Configure** > **Encryption** from the menu task bar to display the **Encryption Center** dialog box (Refer to Figure 6 on page 14).
2. Select an encryption engine from the **Encryption Center Devices** table, then select **Engine** > **Properties** from the menu task bar, or right-click a switch or encryption engine and select **Properties**.

   NOTE
   You can also select a an engine from the **Encryption Center Devices** table, then click the **Targets** icon.

3. In the **Encryption Engine Properties** table, locate **Set State To**.
4. Click the adjacent **Engine** field and select **Enabled** or **Disabled** accordingly, then click **OK**.

Viewing and editing encryption group properties

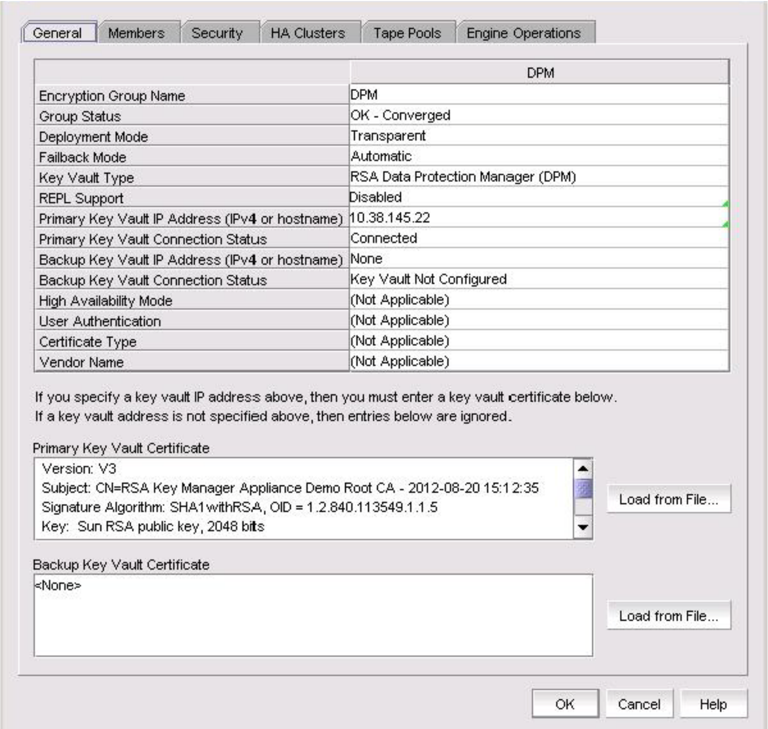
Whenever you add or change a key vault address, you must also load the corresponding key vault certificate. When adding or changing a key vault, if the switches in the encryption group have not been previously registered with the new key vault, you must add the switch certificates to the key vault.

To view encryption group properties, complete the following steps.

1. Select **Configure** > **Encryption** from the menu task bar to display the **Encryption Center** dialog box (Refer to Figure 6 on page 14).
2. Select a group from the **Encryption Center Devices** table, then select **Group** > **Properties** from the menu task bar.
3. You can also select a group from the **Encryption Center Devices** table, then click the **Properties** icon

NOTE
If groups are not visible in the **Encryption Center Devices** table, select **View** > **Groups** from the menu task bar.

The **Encryption Group Properties** dialog box includes several tabs that are used to configure the various functions for encryption groups. All tabs are visible for all key vault types with one exception; the **Link Keys** tab is visible only if the key vault type is NetApp LKM. Unless otherwise specified, the **Encryption Group Properties** dialog box opens with the General tab displayed.



	DPM
Encryption Group Name	DPM
Group Status	OK - Converged
Deployment Mode	Transparent
Failback Mode	Automatic
Key Vault Type	RSA Data Protection Manager (DPM)
REPL Support	Disabled
Primary Key Vault IP Address (IPv4 or hostname)	10.38.145.22
Primary Key Vault Connection Status	Connected
Backup Key Vault IP Address (IPv4 or hostname)	None
Backup Key Vault Connection Status	Key Vault Not Configured
High Availability Mode	(Not Applicable)
User Authentication	(Not Applicable)
Certificate Type	(Not Applicable)
Vendor Name	(Not Applicable)

If you specify a key vault IP address above, then you must enter a key vault certificate below.
If a key vault address is not specified above, then entries below are ignored.

Primary Key Vault Certificate

```
Version: V3
Subject: CN=RSA Key Manager Appliance Demo Root CA - 2012-08-20 15:12:35
Signature Algorithm: SHA1withRSA, OID = 1.2.840.113549.1.1.5
Key:  Sun RSA public key, 2048 bits
```

Backup Key Vault Certificate

```
<None>
```

FIGURE 82 Encryption Group Properties dialog box

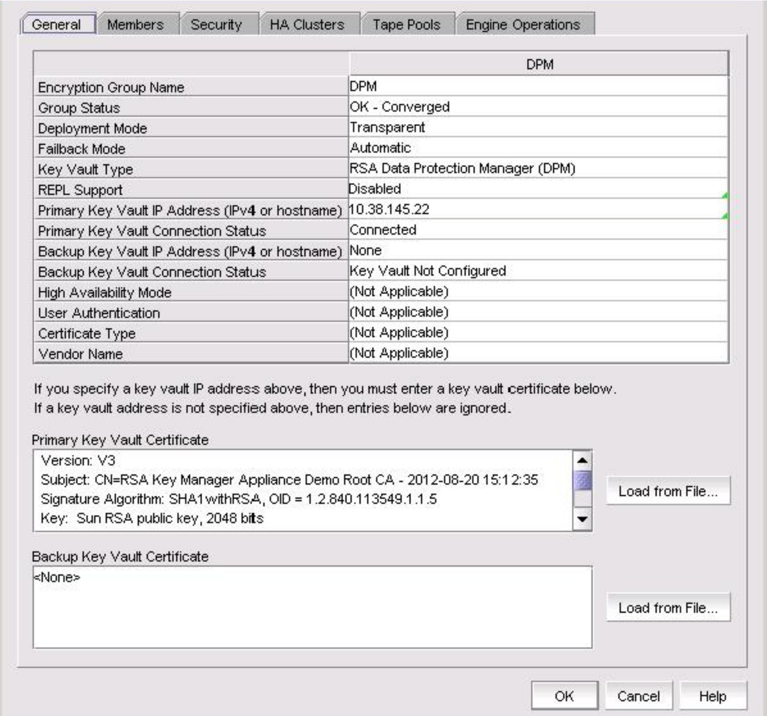
The dialog box contains the following information:

- **General** tab: For a description of the dialog box, refer to "General tab" on page 113.
- **Members** tab: For a description of the dialog box, refer to "Members tab" on page 115.
- **Security** tab: For a description of the dialog box, refer to "Security tab" on page 117.
- **HA Clusters** tab: For a description of the dialog box, refer to "HA Clusters tab" on page 119.
- **Tape Pools** tab: For a description of the dialog box, refer to "Tape Pools tab" on page 121.
- **Engine Operations** tab: For a description of the dialog box, refer to "Engine Operations tab" on page 123.

General tab

The **General** tab (Figure 83) is viewed from the **Encryption Group Properties** dialog box. To access the **General** tab, select a group from the **Encryption Center Devices** table, then select **Group > Properties** from the menu task bar.

NOTE
You can also select a group from the **Encryption Center Devices** table, then click the **Properties** icon.



	DPM
Encryption Group Name	DPM
Group Status	OK - Converged
Deployment Mode	Transparent
Failback Mode	Automatic
Key Vault Type	RSA Data Protection Manager (DPM)
REPL Support	Disabled
Primary Key Vault IP Address (IPv4 or hostname)	10.38.145.22
Primary Key Vault Connection Status	Connected
Backup Key Vault IP Address (IPv4 or hostname)	None
Backup Key Vault Connection Status	Key Vault Not Configured
High Availability Mode	(Not Applicable)
User Authentication	(Not Applicable)
Certificate Type	(Not Applicable)
Vendor Name	(Not Applicable)

If you specify a key vault IP address above, then you must enter a key vault certificate below.
If a key vault address is not specified above, then entries below are ignored.

Primary Key Vault Certificate

```
Version: V3
Subject: CN=RSA Key Manager Appliance Demo Root CA - 2012-08-20 15:12:35
Signature Algorithm: SHA1withRSA, OID = 1.2.840.113549.1.1.5
Key:  Sun RSA public key, 2048 bits
```

Backup Key Vault Certificate

<None>

FIGURE 83 Encryption Group Properties dialog box - General tab

The dialog box contains the following information:

- **Encryption Group Name:** The name of the encryption group
- **Group Status:** The status of the encryption group. Options are:
  - **OK-Converged:** The group leader can communicate with all members
  - **Degraded:** The group leader cannot contact one or more of the configured group members. When the group is in a degraded state, many operations are not permitted, including configuring targets, hosts, LUNs, HA clusters, and tape pools.
- **Deployment Mode:** The group's deployment mode, which is transparent mode
- **Failback Mode:** Identifies the group's failback mode. Options are: Automatic and Manual. Failback mode can be changed by clicking on the field and selecting the desired mode.

  The HA Failback option determines the behavior when a failed encryption engine is restarted. When one encryption engine in an HA cluster fails, the second encryption engine in the HA cluster takes over the encryption and decryption of traffic to all encryption targets in the first encryption engine.

When the first encryption engine comes back online, the encryption group's failback setting determines whether the first encryption engine automatically resumes encrypting and decrypting traffic to its encryption targets. In manual mode, the second encryption engine continues handling the traffic until you manually invoke failback using the CLI, or until the second encryption engine fails.

- **Key Vault Type**: Options are:
  - **RSA Data Protection Manager (DPM): NOTE:** If an encryption group contains mixed firmware nodes, the Encryption Group Properties Key Vault Type name is based on the firmware version of the group leader. For example, If a switch is running Fabric OS 7.1.0 or later, the Key Vault Type is displayed as "RSA Data Protection Manager (DPM)."If a switch is running a Fabric OS version prior to v7.1.0, Key Vault Type is displayed as "RSA Key Manager (RKM)".
- **REPL Support**: Identifies if the remote replication LUN support is enabled or disabled. You can change the current setting by clicking on the field and selecting the desired state.
- **Primary Key Vault IP Address**: The IP address of the primary key vault, either IPv4 or host name.
- **Primary Key Vault Connection Status**: The status of the primary key vault link. In an operating environment, the status should be **Connected**. Other options are:
  - **Unknown/Busy**
  - **Not configured**
  - **Not responding**
  - **Failed authentication**
- **Backup Key Vault IP Address**: *Optional*. The IP address of the backup key vault. This field can be left blank.
- **Backup Key Vault Connection Status**: The status of the backup key vault link. Options are:
  - **Connected**
  - **Unknown/Busy**
  - **Not configured**
  - **Not responding**
  - **Failed authentication**
  - **High Availability Mode:** (*KMIP key vault only.*) Shown as **Not Applicable**.
  - **User Authentication:** (*KMIP key vault only.*) Shown as **Not Applicable**.
  - **Certificate Type:** (*KMIP key vault only.*) Shown as **Not Applicable**.
  - **Vendor Name:** (*KMIP key vault only.*) Shown as **Not Applicable**.
  - **Primary Key Vault Certificate** table: Displays the details of the primary vault certificate; for example, version and signature information. The **Load from File** button allows you to locate and load a primary key vault certificate from a different location.
  - **Backup Key Vault Certificate** table: Displays the details of the backup vault certificate; for example, version and signature information. The **Load from File** button allows you to locate and load a backup key vault certificate from a different location.

Members tab

The **Members** tab lists group switches, their role, and their connection status with the group leader. The table columns are not editable. The tab displays the configured membership for the group and includes the following:

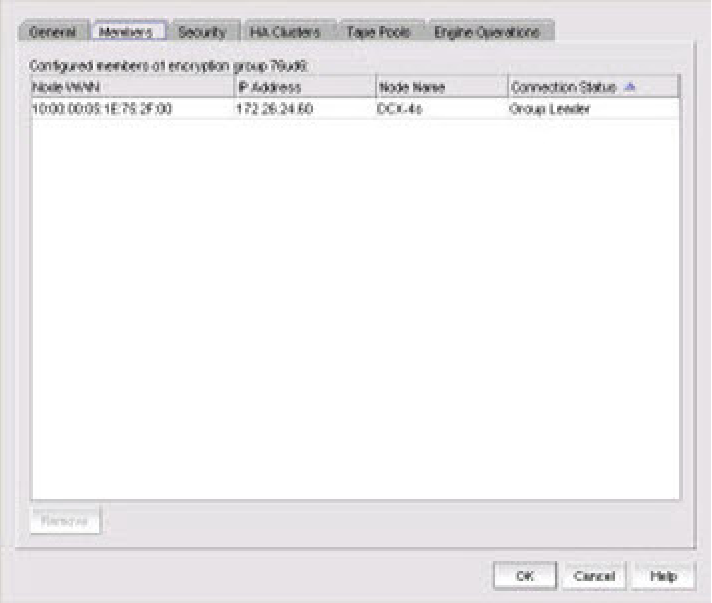
- **Node WWN:** The member switch's world wide name.
- **IP Address:** The switch's IP address or host name.
- **Node Name:** The switch's node name, if known. If unknown, this field is blank.
- **Connection Status:** The switch's connection status. Possible values are:
  - **Goup Leader:** The switch designated as the group leader, so there is no connection status.
  - **Trying to Contact:** The member is not responding to the group leader. This might occur if the member switch is not reachable by way of the management port, or if the member switch does not believe it is part of the encryption group.
  - **Configuring:** The member switch has responded and the group leader is exchanging information. This is a transient condition that exists for a short time after a switch is added or restored to a group.
  - **OK:** The member switch is responding to the group leader switch.
  - **Not Available:** The group leader is not a managed switch, so connection statuses are not being collected from the group leader.

The **Members** table might not match the list of members displayed in the **Encryption Center** dialog box if some configured members are unmanaged, missing, or in a different group.

NOTE
When the encryption group is in the Degraded state, the **Members** tab indicates the group member that the leader cannot contact. If the non-responding switch should no longer be included in the encryption group, it can be removed using the **Remove** button.

The **Members** tab (Figure 84) is viewed from the **Encryption Group Properties** dialog box. To access the **Members** tab, select a group from the **Encryption Center Devices** table, then select **Group > Properties** from the menu task bar.

NOTE
You can also select a group from the **Encryption Center Devices** table, then click the **Properties** icon.



Configured members of encryption group 76ud6:

Node WWN	IP Address	Node Name	Connection Status ▲
10:00:00:05:1E:75:2F:00	172.26.24.60	DCX-4s	Group Leader

FIGURE 84 Encryption Group Properties dialog box - Members tab

Members tab Remove button

You can click the **Remove** button to remove a selected switch or group from the encryption group table.

- You cannot remove the group leader unless it is the only switch in the group. If you remove the group leader, BNA also removes the HA cluster, the target container, and the tape pool (if configured) that are associated with the switch.
- If you remove a switch from an encryption group, BNA also removes the HA cluster and target container associated with the switch.

NOTE
If the encryption group is in a degraded state, BNA does not remove the HA clusters or target containers associated with the switch. In this case, a pop-up error message displays.

- If you remove the last switch from a group, BNA also deletes the group.

Consequences of removing an encryption switch

The consequences of removing a switch from an encryption group are as follows:

- All configured targets on the switch are deleted from the switch's configuration.
- Any encryption being performed by the switch is halted.
- If the removed switch was in an HA cluster, the switch can no longer provide HA support. HA clusters that contained the encryption engine from the removed switch are deleted.

The consequences of removing the last switch in a group (which will be the group leader) are all switch removal consequences noted above, plus the following:

- The encryption group is deleted.
- All configured tape pools are deleted.

Table 2 explains the impact of removing switches.

TABLE 2 Switch removal impact

Switch configuration	Impact of removal
The switch is the only switch in the encryption group.	The encryption group is also removed.
The switch has configured encryption targets on encryption engines.	• The switch is configured to encrypt traffic to one or more encryption targets. • The target container configuration is removed. • The encrypted data remains on the encryption target but is not usable until the encryption target is manually configured on another encryption switch. **CAUTION** **The encryption target data is visible in encrypted format to zoned hosts. It is strongly recommended that you remove the encryption targets from all zones before you disable encryption. Otherwise, hosts might corrupt the encrypted data by writing directly to the encryption target without encryption.**
The switch has encryption engines in HA Clusters.	The HA Clusters are removed. High availability is no longer provided to the other encryption engine in each HA Cluster.

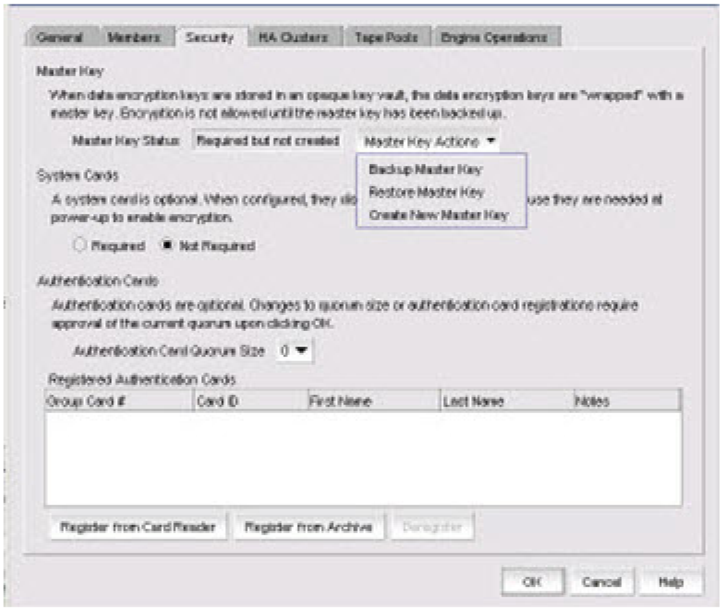
A warning message is displayed when you attempt to remove a switch or an encryption group. After you have read the warning, you must click **Yes** to proceed.

Security tab

The **Security** tab displays the status of the master key for the encryption group and whether smart cards are required. From here, you register smart cards for use.

The **Security** tab (Figure 85) is viewed from the **Encryption Group Properties** dialog box. To access the **Security** tab, select a group from the **Encryption Center Devices** table, then select **Group > Security** from the menu task bar. The **Properties** dialog box displays with the **Security** tab selected.

NOTE
You can also select a group from the **Encryption Center Devices** table, then click the **Properties** icon.



Master Key

When data encryption keys are stored in an opaque key vault, the data encryption keys are "wrapped" with a master key. Encryption is not allowed until the master key has been backed up.

Master Key Status: Required but not created

System Cards

A system card is optional. When configured, they slo ... use they are needed at power-up to enable encryption.

○ Required ● Not Required

Authentication Cards

Authentication cards are optional. Changes to quorum size or authentication card registrations require approval of the current quorum upon clicking OK.

Authentication Card Quorum Size: 0

Registered Authentication Cards

Group Card #	Card ID	First Name	Last Name	Notes

FIGURE 85 **Encryption Group Properties dialog box - Security tab**

The dialog box contains the following information:

- Master Key Status: Displays the status of the master key. Possible values are:
  - Not used: Displays when LKM is the key vault.
  - Required but not created: Displays when a master key needs to be created.
  - Created but not backed up: Displays when the master key needs to be backed up. For safety, the master key cannot be used until it is backed up.
  - Created and backed up: Indicates the master key is usable.
- Master Key Actions list: Master Key actions are disabled if the master key state is not correct. Master key actions are:
  - Create a new master key: Enabled when no master key exists or the previous master key has been backed up.
  - Back up a master key: Enabled any time a master key exists.
  - Restore a master key: Enabled when either no master key exists or the previous master key has been backed up.
- System Cards: Identifies if the use of a system card is required for controlling activation of the encryption engine. You must indicate if cards are required or not required. If a system card is required, it must be read by the card reader on the switch.
- Authentication Cards, which identifies if one or more authentication cards must be read by a card reader attached to a Management application PC to enable certain security-sensitive operations.
- Authentication Cards quorum size selector: Determines the number of registered authentication cards needed for a quorum. The number should always be one less than the actual number registered.

NOTE
When registering authentication cards, you must register the defined quorum size plus one.

- **Registered Authentication Cards** table: Lists the registered authentication cards by Group Card number, Card ID, the name of the person to which the card is assigned, and optional notes.
- **Register from Card Reader** button: Launches the **Add Authentication Card** dialog box.
- **Register from Archive** button: Launches the **Add Authentication Card** dialog box.
- **Deregister** button: Deregisters authentication cards, thus enabling them to be removed from the switch and the database.

Encryption is not allowed until the master key has been backed up. Master keys are needed for all key vaults except LKM.

NOTE
You must enable encryption engines before you back up or restore master keys.

NOTE
If all encryption engines are otherwise okay but are missing the master key, the following message displays below the Master Key status:

```
"None of the encryption engines in this encryption group have a copy of the master
key. The master key should be restored from a backup."
```

This situation can occur if all encryption engines in a group are zeroized and then re-enabled.

HA Clusters tab

The **HA Clusters** tab allows you to create and delete HA clusters, add encryption engines to and remove encryption engines from HA clusters, and failback an engine. Changes are not applied to the encryption group until you click **OK**.

Each HA Cluster must have exactly two encryption engines. The two encryption engines in the cluster must be in the same fabric (they will always be in the same encryption group since only the engines in the group are listed for selection).

HA clusters are groups of encryption engines that provide high availability features. If one of the engines in the group fails or becomes unreachable, the other cluster member takes over the encryption and decryption tasks of the failed encryption engine. An HA cluster consists of exactly two encryption engines. See "Creating HA clusters" on page 52.

The **HA Clusters** tab (Figure 86) is viewed from the **Encryption Group Properties** dialog box. To access the **HA Clusters** tab, select a group from the **Encryption Center Devices** table, then select **Group > HA Clusters** from the menu task bar. The **Properties** dialog box displays with the **HA Clusters** tab selected.

NOTE
You can also select a group from the **Encryption Center Devices** table, then click the **Properties** icon.

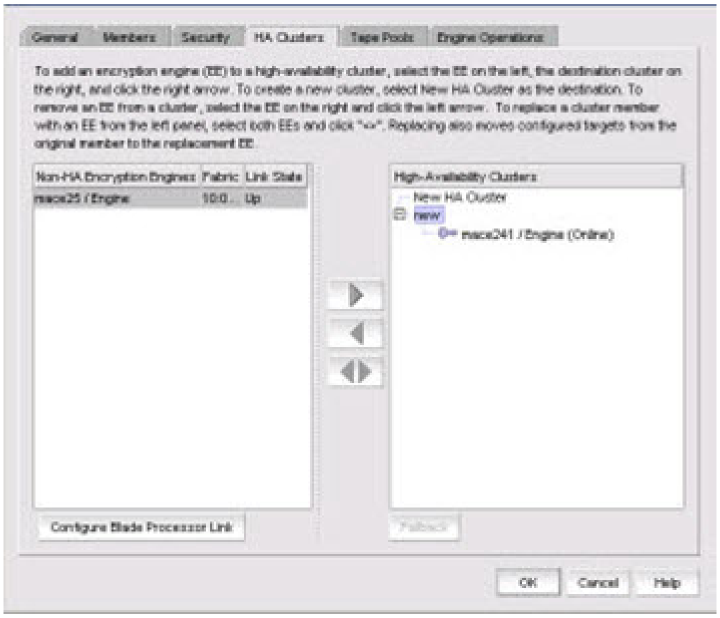
The tab displays the includes the following information:

- **Non-HA Encryption Engines** table: Displays a list of encryption engines that are not configured for high-availability clustering
- **High-Availability Clusters** table: A list of encryption engines that have been selected for high-availability clustering.

- **Right**- and **Left-arrow** buttons: You can select an encryption engine in the **Non-HA Encryption Engines** table and click the **Right-arrow** button to add the encryption engine to the **High-Availability Clusters.** (If you are creating a new HA cluster, a dialog box displays requesting a name for the new HA cluster.) Similarly, you can select an encryption engine in the **High-Availability Clusters** table and click the **Left-arrow** button to remove it from a cluster. The encryption engine is removed from the table and shown as available.
- **Dual-arrow** buttons: After selecting an encryption engine in both the **Non-HA Encryption Engines** table and the **High-Availability Clusters** table, clicking the **Dual-arrow** button swaps the cluster members.

NOTE
Swapping engines using the **Dual-arrow** button is not the same as removing one engine and adding another. When swapping engines, all configured targets are moved from the former HA Cluster member to the new HA Cluster member. Swapping engines is useful when replacing hardware.

- **Configure Blade Processor Link** button: When active, clicking the button displays the Configure Blade Processor Link dialog box. Blade processor links must be configured and functioning to enable the failover/failback capabilities of a high availability cluster. For more information, refer to "Configuring blade processor links" on page 28.
- **Failback** button: After selecting an online encryption engine in the **High-Availability Clusters** table, you can click **Failback** to manually invoke failback. For more information, refer to "Invoking failback" on page 54.



To add an encryption engine (EE) to a high-availability cluster, select the EE on the left, the destination cluster on the right, and click the right arrow. To create a new cluster, select New HA Cluster as the destination. To remove an EE from a cluster, select the EE on the right and click the left arrow. To replace a cluster member with an EE from the left panel, select both EEs and click "<>". Replacing also moves configured targets from the original member to the replacement EE.

Non-HA Encryption Engines	Fabric	Link State
mace25 / Engine	10:0...	Up

High-Availability Clusters
- New HA Cluster
- new
  - mace241 / Engine (Online)

FIGURE 86 Encryption Group Properties dialog box - HA Clusters tab

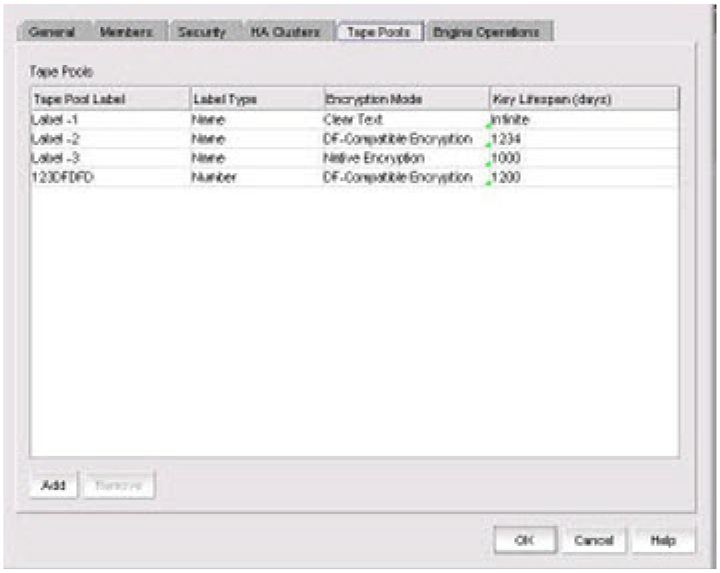
Tape Pools tab

Tape pools are managed from the **Tape Pools** tab. From the **Tape Pools** tab, you can add, modify, and remove tape pools.

- To add a tape pool, click **Add**, then complete the **Add Tape Pool** dialog box.
- To remove an encryption switch or engine from a tape pool, select one or more tape pools listed in the table, then click **Remove**.
- To modify a tape pool, you must remove the entry, then add a new tape pool.

The **Tape Pools** tab (Figure 87) is viewed from the **Encryption Group Properties** dialog box. To access the **Tape Pools** tab, select a group from the **Encryption Center Devices** table, then select **Group** > **Tape Pools** from the menu task bar. The **Properties** dialog box displays with the **Tape Pools** tab selected.

NOTE
You can also select a group from the **Encryption Center Devices** table, then click the **Properties** icon.



Tape Pools

Tape Pool Label	Label Type	Encryption Mode	Key Lifespan (days)
Label -1	Name	Clear Text	Infinite
Label -2	Name	DF-Compatible Encryption	1234
Label -3	Name	Native Encryption	1000
123DFDFD	Number	DF-Compatible Encryption	1200

FIGURE 87 **Encryption Group Properties dialog box - Tape Pools tab**

Tape pools overview

Tape cartridges and volumes can be organized into a tape pool (a collection of tape media). The same data encryption keys are used for all cartridges and volumes in the pool. Tape pools are used by backup application programs to group all tape volumes used in a single backup or in a backup plan. The tape pool name or number used must be the same name or number used by the host backup application. If the same tape pool name or number is configured for an encryption group, tapes in that tape pool are encrypted according to the tape pool settings instead of the tape LUN settings.

Encryption switches and encryption blades support tape encryption at the tape pool level (for most backup applications) and at the LUN (tape drive) level. Since Tape Pool policies override the LUN (tape drive) policies, the LUN pool policies are used only if no tape pools exist or if the tape media/volume does not belong to any configured tape pools.

All encryption engines in the encryption group share the tape pool definitions. Tapes can be encrypted by any encryption engine in the group where the container for the tape target LUN is hosted. The tape media is mounted on the tape target LUN.

Tape pool definitions are not needed to read a tape. The tape contains enough information (encryption method and key ID) to read the tape. Tape pool definitions are only used when writing to tape. Tape pool names and numbers must be unique within the encryption group.

Adding tape pools

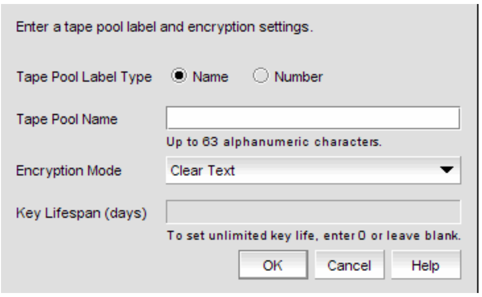
A tape pool can be identified by either a name or a number, but not both. Tape pool names and numbers must be unique within the encryption group. When a new encryption group is created, any existing tape pools in the switch are removed and must be added.

1. Select **Configure** > **Encryption** from the menu task bar to display the **Encryption Center** dialog box (Refer to Figure 6 on page 14).
2. Select a group from the **Encryption Center Devices** table, then select **Group** > **Tape Pools** from the menu task bar.

> **NOTE**
> If groups are not visible in the **Encryption Center Devices** table, select **View** > **Groups** from the menu task bar.

3. Click **Add**.

   The **Add Tape Pool** dialog box displays (Figure 88). The **Name** tape pool label type is the default; however, you can change the tape pool label type to **Number** (Figure 89).


Enter a tape pool label and encryption settings.

Tape Pool Label Type ● Name ○ Number

Tape Pool Name

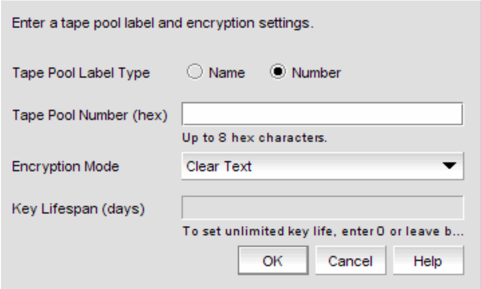
Up to 63 alphanumeric characters.

Encryption Mode: Clear Text

Key Lifespan (days)

To set unlimited key life, enter 0 or leave blank.

FIGURE 88 Add Tape Pool by name dialog box


Enter a tape pool label and encryption settings.

Tape Pool Label Type ○ Name ● Number

Tape Pool Number (hex)

Up to 8 hex characters.

Encryption Mode: Clear Text

Key Lifespan (days)

To set unlimited key life, enter 0 or leave b...

FIGURE 89 Add Tape Pool by number dialog box

4. Based on your selection, do one of the following:
   - If you selected **Name** as the **Tape Pool Label Type**, enter a name for the tape pool. This name must match the tape pool label or tape ID that is configured on the tape backup/restore application.
   - If you selected **Number** as the **Tape Pool Label Type**, enter a (hex) number for the tape pool. This number must match the tape pool label or tape number that is configured on the tape backup/restore application.
5. Select the **Encryption Mode**. Options are **Clear Text,** and **Native Encryption**. Note the following:
   - The **Key Lifespan (days)** field is editable only if the tape pool is encrypted.
   - If **Clear Text** is selected as the encryption mode, the key lifespan is disabled.

   NOTE
   You cannot change the encryption mode after the tape pool I/O begins.

6. Enter the number of days to use a key before obtaining a new one, if you choose to enforce a key lifespan. The default is Infinite (a blank field or a value of 0), which is the recommended setting.

   NOTE
   The key lifespan interval represents the key expiry timeout period for tapes or tape pools. You can only enter the **Key Lifespan** field if the tape pool is encrypted. If Clear Text is selected as the encryption mode, the **Key Lifespan** field is disabled.

7. Click **OK**.

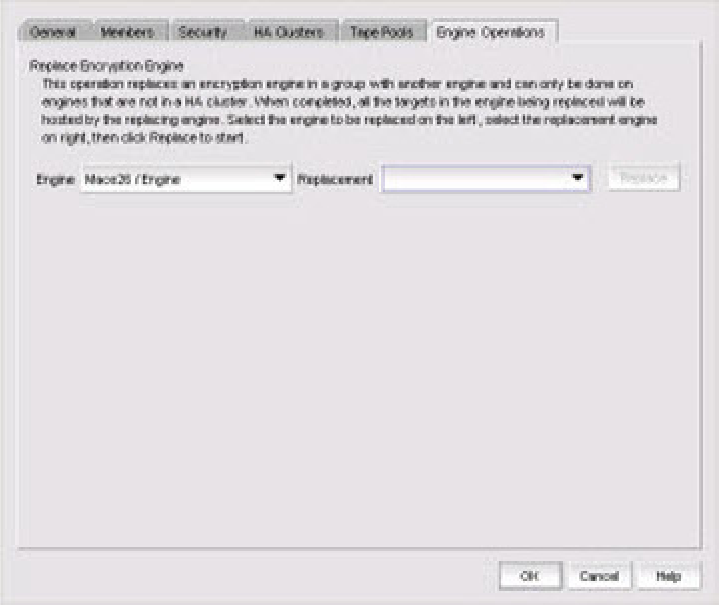
Engine Operations tab

The **Engine Operations** tab enables you to replace an encryption engine in a switch with another encryption engine in another switch within a DEK Cluster environment. A DEK Cluster is a set of encryption engines that encrypt the same target storage device. DEK Clusters do not display in BNA; they are an internal implementation feature and have no user-configurable properties. Refer to "Replacing an encryption engine in an encryption group" on page 50.

The **Engine Operations** tab (Figure 86) is viewed from the **Encryption Group Properties** dialog box. To access the **Engine Operations** tab, select a group from the **Encryption Center Devices** table, then select **Group > Engine Operations** from the menu task bar. The **Properties** dialog box displays with the **Engine Operations** tab selected.

NOTE
You can also select a group from the **Encryption Center Devices** table, then click the **Properties** icon.

You simply select the encryption engine you want to replace from the Engine list, select the encryption engine to use for the group from the **Replacement** list, then click **Replace.**



Replace Encryption Engine

This operation replaces an encryption engine in a group with another engine and can only be done on engines that are not in a HA cluster. When completed, all the targets in the engine being replaced will be hosted by the replacing engine. Select the engine to be replaced on the left, select the replacement engine on right, then click Replace to start.

Engine: Macs26 / Engine

Replacement:

FIGURE 90 Encryption Group Properties Dialog Box - Engine Operations Tab

NOTE
You cannot replace an encryption engine if it is part of an HA Cluster.

Encryption-related acronyms in log messages

Fabric OS log messages related to encryption components and features may have acronyms embedded that require interpretation. Table 3 lists some of those acronyms.

TABLE 3 Encryption acronyms

Acronym	Name
EE	Encryption Engine
EG	Encryption Group
HAC	High Availability Cluster

Chapter

Configuring Encryption Using the CLI

3

In this chapter

Overview

This chapter explains how to use the command line interface (CLI) to configure a Brocade Encryption Switch, or an FS8-18 Encryption blade in a DCX Backbone chassis to perform data encryption.

This chapter assumes that the basic setup and configuration of the Brocade Encryption Switch, and DCX Backbone chassis have been done as part of the initial hardware installation, including setting the management port IP address.

For command syntax and description of parameters, refer to the *Fabric OS Command Reference Manual.*

Command validation checks

Before a command is executed, it is validated against the following checks.

1. **Active or Standby availability:** on enterprise-class platforms, checks that the command is available on the Control Processor (CP).
2. **Role Based Access Control (RBAC) availability:** checks that the invoking user's role is permitted to invoke the command. If the command modifies system state, the user's role must have *modify* permission for the command. If the command only displays system state, the user's role must have *observe* permission for the command. Some commands both observe and modify system state and thus require *observe-modify* permission. The following RBAC permissions are supported:
   - O = observe
   - OM = observe-modify
   - N = none/not available
3. **Admin Domain availability:** checks that the command is allowed in the currently selected Admin Domain. For information on Admin Domain concepts and restrictions, refer to the g126
4. .

   Admin Domain Types are one or more of the following. If more than one AD type is listed for a command, the AD type is option-specific. Display options may be allowed, but set options may be subject to Admin Domain restrictions.

SwitchMember	Allowed to execute only if the local switch is part of the current AD.
Allowed	Allowed to execute in all ADs.
PhysFabricOnly	Allowed to execute only in AD255 context (and the user should own access to AD0-AD255 and have admin RBAC privilege).
Disallowed	Allowed to execute in AD0 or AD255 context only; not allowed in AD1-AD254 context.
AD0Disallowed	Allowed to execute only in AD255 and AD0 (if no ADs are configured).
AD0Only	Allowed to execute only in AD0 when ADs are not configured.
Command-specific	Checks whether the command is supported on the platform for which it is targeted.

5. PortMember: allows all control operations only if the port or the local switch is part of the current AD. View access is allowed if the device attached to the port is part of the current AD.

Command RBAC permissions and AD types

Two RBAC roles are permitted to perform Encryption operations.

- Admin and SecurityAdmin

  Users authenticated with the *Admin and SecurityAdmin RBAC* roles may perform cryptographic functions assigned to the FIPS Crypto Officer, including the following:

  - Perform encryption node initialization.
  - Enable cryptographic operations.
  - Manage I/O functions for critical security parameters (CSPs).
  - Zeroize encryption CSPs.
  - Register and configure a key vault.
  - Configure a recovery share policy.
  - Create and register recovery share.
  - Perform encryption group- and clustering-related operations.
  - Manage keys, including creation, recovery, and archive functions.

- Admin and FabricAdmin

  Users authenticated with the *Admin and FabricAdmin RBAC* roles may perform routine Encryption Switch management functions, including the following:

  - Configure virtual devices and crypto LUNs.
  - Configure LUN and tape associations.
  - Perform rekeying operations.
  - Perform firmware download.
  - Perform regular Fabric OS management functions.

See Table 4 for the RBAC permissions when using the encryption configuration commands.

TABLE 4 Encryption command RBAC availability and admin domain type[1]

Command name	User	Admin	Operator	Switch Admin	Zone Admin	Fabric Admin	Basic Switch Admin	Security Admin	Admin Domain
addmembernode	N	OM	N	N	N	N	N	OM	Disallowed
addhaclustermember	N	OM	N	N	N	OM	N	N	Disallowed
addinitiator	N	OM	N	N	N	OM	N	N	Disallowed
addLUN	N	OM	N	N	N	OM	N	N	Disallowed
commit	N	OM	N	N	N	OM	N	N	Disallowed
createcontainer	N	OM	N	N	N	OM	N	N	Disallowed
createencgroup	N	OM	N	N	N	N	N	OM	Disallowed

TABLE 4 Encryption command RBAC availability and admin domain type[1] (Continued)

Command name	User	Admin	Operator	Switch Admin	Zone Admin	Fabric Admin	Basic Switch Admin	Security Admin	Admin Domain
createhacluster	N	OM	N	N	N	OM	N	N	Disallowed
createtapepool	N	OM	N	N	N	OM	N	N	Disallowed
decommission	N	OM	N	N	N	OM	N	N	Disallowed
deletecontainer	N	OM	N	N	N	OM	N	N	Disallowed
deletedecommissionedkeyids	N	OM	N	N	N	OM	N	N	Disallowed
deleteencgroup	N	OM	N	N	N	N	N	OM	Disallowed
deletefile	N	OM	N	N	N	N	N	OM	Disallowed
deletehacluster	N	OM	N	N	N	OM	N	N	Disallowed
deletetapepool	N	OM	N	N	N	OM	N	N	Disallowed
deregkeyvault	N	OM	N	N	N	N	N	OM	Disallowed
deregmembernode	N	OM	N	N	N	N	N	OM	Disallowed
disableEE	N	OM	N	N	N	N	N	OM	Disallowed
discoverLUN	N	OM	N	N	N	OM	N	N	Disallowed
eject	N	OM	N	N	N	N	N	OM	Disallowed
enable	N	OM	N	N	N	OM	N	N	Disallowed
enableEE	N	OM	N	N	N	N	N	OM	Disallowed
export	N	OM	N	N	N	N	N	OM	Disallowed
exportmasterkey	N	OM	N	N	N	N	N	OM	Disallowed
failback	N	OM	N	N	N	OM	N	N	Disallowed
genmasterkey	N	OM	N	N	N	N	N	OM	Disallowed
help	N	OM	N	N	N	OM	N	OM	Disallowed
import	N	OM	N	N	N	N	N	OM	Disallowed
initEE	N	OM	N	N	N	N	N	OM	Disallowed
initnode	N	OM	N	N	N	N	N	OM	Disallowed
kvdiag	N	OM	N	N	N	N	N	OM	Disallowed
leave_encryption_group	N	OM	N	N	N	N	N	OM	Disallowed
manual_rekey	N	OM	N	N	N	OM	N	N	Disallowed
modify	N	OM	N	N	N	OM	N	N	Disallowed
move	N	OM	N	N	N	OM	N	N	Disallowed
perfshow	N	OM	N	N	N	OM	N	O	Disallowed

TABLE 4 Encryption command RBAC availability and admin domain type[1] (Continued)

Command name	User	Admin	Operator	Switch Admin	Zone Admin	Fabric Admin	Basic Switch Admin	Security Admin	Admin Domain
rebalance	N	OM	N	N	N	OM	N	N	Disallowed
reclaim	N	OM	N	N	N	OM	N	N	Disallowed
recovermasterkey	N	OM	N	N	N	N	N	OM	Disallowed
refreshdek	N	OM	N	N	N	N	N	OM	Disallowed
regEE	N	OM	N	N	N	N	N	OM	Disallowed
regKACcert	N	OM	N	N	N	N	N	OM	Disallowed
regKAClogin	N	OM	N	N	N	N	N	OM	Disallowed
regkeyvault	N	OM	N	N	N	N	N	OM	Disallowed
regmembernode	N	OM	N	N	N	N	N	OM	Disallowed
removehaclustermember	N	OM	N	N	N	OM	N	N	Disallowed
removeinitiator	N	OM	N	N	N	OM	N	N	Disallowed
removeLUN	N	OM	N	N	N	OM	N	N	Disallowed
replace	N	OM	N	N	N	OM	N	N	Disallowed
resume_rekey	N	OM	N	N	N	OM	N	N	Disallowed
set	N	OM	N	N	N	N	N	OM	Disallowed
show	N	OM	N	N	N	O	N	OM	Disallowed
transabort	N	OM	N	N	N	OM	N	N	Disallowed
transshow	N	OM	N	N	N	OM	N	O	Disallowed
zeroizeEE	N	OM	N	N	N	N	N	OM	Disallowed

[1] Legend: O = observe, OM = observe-modify, N = none/not available

Cryptocfg Help command output

All encryption operations are done using the **cryptocfg** command. The **cryptocfg** command has a help output that lists all options.

```
switch:admin> cryptocfg --help
Usage: cryptocfg
--help -nodecfg:
        Display the synopsis of node parameter configuration.
--help -groupcfg:
        Display the synopsis of group parameter configuration.
--help -hacluster:
        Display the synopsis of hacluster parameter configuration.
--help -devicecfg:
        Display the synopsis of device container parameter configuration.
--help -transcfg:
        Display the synopsis of transaction management.

switch:admin> cryptocfg --help -nodecfg
Usage: cryptocfg
--help -nodecfg:
        Display the synopsis of node parameter configuration.
--initnode:
        Initialize the node for configuration of encryption options.
--initEE [<slotnumber>]:
        Initialize the specified encryption engine.
--regEE [<slotnumber>]:
        Register a previously initialized encryption blade.
--reg -membernode <member node WWN> <member node certfile> <IP addr>:
        Register a member node with the system.
--reg -groupleader <group leader WWN> <group leader certfile> <IP addr>:
        Register a group leader node with the system.
(output truncated)
```

Management LAN configuration

Each encryption switch has one GbE management port. In the case of a DCX Backbone chassis with FS8-18 blades installed, management ports are located on the CP blades. The management port IP address is normally set as part of the hardware installation. A static IP address should be assigned. To eliminate DNS traffic and potential security risks related to DHCP, DHCP should not be used.

For encryption switches and blades, the management port is used to communicate with a key management system, and a secure connection must be established between the management port and the key management system. All switches you plan to include in an encryption group must be connected to the key management system. Only IPv4 addressing is currently supported. All nodes, including the key management system, must use the same version of IP addressing.

Configuring cluster links

Each encryption switch or FS8-18 blade has two gigabit Ethernet ports labeled Ge0 and Ge1. The Ge0 and Ge1 ports connect encryption switches and FS8-18 blades to other encryption switches and FS8-18 blades. These two ports are bonded together as a single virtual network interface. Only one IP address is used. The ports provide link layer redundancy, and are collectively referred to as the cluster link.

NOTE
Do not confuse the gigabit Ethernet ports with the management and console ports, which are also RJ-45 ports located close to the gigabit Ethernet ports.

All encryption switches or blades in an encryption group must be interconnected by their cluster links through a dedicated LAN. Both ports of each encryption switch or blade must be connected to the same IP network and the same subnet. Static IP addresses should be assigned. Neither VLANs nor DHCP should be used.

1. Log in to the switch as Admin or FabricAdmin.
2. Configure the IP address using the **ipAddrSet** command. Only Ge0 needs to be configured. Always use **ipAddrSet –eth0** to configure the address. If an address is assigned to ge1 (-eth1), it is accepted and stored, but it is ignored. Only IPv4 addresses are supported for cluster links.

   The following example configures a static IP address and gateway address for the bonded interface.

   ```
   switch:admin> ipaddrset -eth0 --add 10.32.33.34/23
   switch:admin> ipaddrset -gate --add 10.32.1.1
   ```

Special consideration for blades

HA clusters of FS8-18 blades should not include blades in the same DCX Backbone chassis.

For FS8-18 blades, the slot number must also be included in the **ipAddrSet** command, for example:

```
switch:admin> ipaddrset -slot 7 -eth0 --add 10.32.33.34/23
switch:admin> ipaddrset -slot 7 -gate --add 10.32.1.1
```

There are additional considerations if blades are removed and replaced, or moved to a different slot. On chassis-based systems, IP addresses are assigned to the slot rather than the blade, and are saved in non-volatile storage on the control processor blades. IP addresses may be assigned even if no blade is present. If an FS8-18 blade is installed in a slot that was previously configured for a different type of blade with two IP ports (an FC4-16E blade, for example), the FS8-18 blade is assigned the address specified for -eth0 in that slot.

To be sure the correct IP addresses are assigned, use the **ipAddrShow** command to display the IP address assignments, as shown in the following example:

```
switch:admin> ipaddrshow -slot 7

SWITCH
Ethernet IP Address: 10.33.54.207
Ethernet Subnetmask: 255.255.240.0
Fibre Channel IP Address: none
Fibre Channel Subnetmask: none
Gateway IP Address: 10.33.48.1
```

```
DHCP: Off
eth0: 10.33.54.208/20
eth1: none/none
Gateway: 10.33.48.1
```

NOTE
The IP address of the cluster link should be configured before enabling the encryption engine for encryption. If the IP address is configured after the encryption engine is enabled for encryption, or if the IP address of the cluster link ports is modified after the encryption engine is enabled for encryption, the encryption switch must be rebooted, and the encryption blade must be powered off and powered on (slotpoweroff/slotpoweron) for the IP address configuration to take effect. Failure to do so will result in the rekey operation not starting in the encryption group or high availability (HA) cluster.

IP Address change of a node within an encryption group

Modifying the IP address of a node that is part of an encryption group is disruptive in terms of cluster operation. The change causes the encryption group to split, and if the node was part of an HA cluster, failover/failback capability is lost. The **ipAddrSet** command issues no warning and you are not prevented from changing a node IP address that is part of a configured encryption group or HA cluster. The recommended steps for modifying the IP address of a node are provided below. the procedures are based on whether the node is a group leader or a member node.

Node is a group leader node

1. Log in to the group leader as Admin or SecurityAdmin.
2. Reboot the encryption switch/DCX Backbone chassis (both active and standby central processors) so the existing group leader fails over and one of the member nodes assumes the role of group leader.
   a. If the Encryption Group (EG) is not a single node EG, reboot the encryption switch/DCX Backbone chassis (both active and standby central processors) so the existing group leader fails over and one of the member nodes assumes the role of group leader.
   b. If the node is a single node EG, complete the following steps:
      1. Delete the encryption group.
      2. Change the IP of the switch.
      3. Create the encryption group.
3. After the encryption group is converged, complete the steps noted in "Node is a member node".

Node is a member node

1. Log in to the group leader as Admin or SecurityAdmin.
2. Eject and deregister the node from the encryption group.
3. Change the IP address of the member node using the new IP address.
4. Reboot the member node (the node on which the IP address has been modified).

5. Reregister the node with the group leader using new IP address.

Setting encryption node initialization

When an encryption node is initialized, the following security parameters and certificates are generated:

- FIPS crypto officer
- FIPS user
- Node CP certificate
- A signed Key Authentication Center (KAC) certificate
- A KAC Certificate Signing Request (CSR)

From the standpoint of external SAN management application operations, the FIPS crypto officer, FIPS user, and node CP certificates are transparent to users. The KAC certificates are required for operations with key managers. In most cases, KAC certificate signing requests must be sent to a Certificate Authority (CA) for signing to provide authentication before the certificate can be used. In all cases, signed KACs must be present on each switch.

1. Initialize the Brocade Encryption Switch node.

```
SecurityAdmin:switch> cryptocfg --initnode
Operation succeeded.
```

2. Initialize the new encryption engine.

```
SecurityAdmin:switch> cryptocfg --initEE [slotnumber]
Operation succeeded.
```

3. Register the encryption engine.

```
SecurityAdmin:switch> cryptocfg --regEE [slotnumber]
Operation succeeded.
```

4. Enable the encryption engine.

```
SecurityAdmin:switch> cryptocfg --enableEE [slotnumber]
Operation succeeded.
```

5. Check the encryption engine state using following command to ensure encryption engine is online:

```
SecurityAdmin:switch> cryptocfg --show -localEE
```

Steps for connecting to a DPM appliance

All switches you plan to include in an encryption group must have a secure connection to the Data Protection Manager (DPM). The following procedure is a suggested order of steps for creating a secure connection to DPM.

NOTE
The Brocade Encryption Switch will not use the Identity Auto Enrollment feature supported with DPM 3.x servers. You must complete the identity enrollment manually to configure the DPM 3.x server with the Brocade Encryption Switch. Refer to "Client registration for manual enrollment" on page 140.

1. Initialize the encryption engines on every Fabric OS encryption node that is expected to perform encryption within the fabric. The **cryptocfg --initnode** command generates a Key Archive Client Certificate Signing Request (KAC CSR) that must be present to enable subsequent steps. Refer to "Initializing the Fabric OS encryption engines" on page 135.
2. Export the KAC CSR to a location accessible to a certificate authority (CA) for signing. Refer to "Exporting the KAC certificate signing request (CSR)" on page 136.
3. Submit the KAC CSR for signing by a CA. Refer to "Submitting the CSR to a CA" on page 136.
4. Import the signed certificate into the Fabric OS encryption node. Refer to "Importing the signed KAC certificate" on page 137.
5. Upload the CA certificate onto the DPM key vault. Refer to "Uploading the CA certificate onto the DPM appliance (and first-time configurations)" on page 138.
6. Upload the KAC certificate onto the DPM appliance, then select the appropriate key classes. Refer to "Uploading the KAC certificate onto the DPM apliance (manual identity enrollment)" on page 139.
7. If dual DPM appliances are used for high availability, the DPM appliances must be clustered and must operate in maximum availability mode, as described in the DPM appliance user documentation.
8. Create a Brocade encryption group. Refer to "Creating a Brocade encryption group" on page 139.
9. Register the DPM on the group leader by exporting the CA certificate for the CA that signed the DPM certificate. Refer to "Client registration for manual enrollment" on page 140.

NOTE
DPM is formerly referred to as RKM. DPM 3.x servers are referred to as DPM. DPM is compatible with Fabric OS 7.1.0 and later. RSA servers using the RKM 2.1.1 client are compatible with earlier Fabric OS versions (for example, v7.0.1) are still referred to as RKM.

Initializing the Fabric OS encryption engines

You must perform a series of encryption engine initialization steps on every Fabric OS encryption node (switch or blade) that is expected to perform encryption within the fabric.

NOTE
The initialization process overwrites any authentication data and certificates that reside on the node and the security processor. If this is not a first-time initialization, make sure to export the master key by running **cryptocfg --exportmasterkey** and **cryptocfg --export -scp --currentMK** before running **--initEE**.

To initialize an encryption engine, complete the following steps:

1. Log in to the switch as Admin or SecurityAdmin.
2. Zeroize all critical security parameters (CSPs) on the switch by entering the **cryptocfg --zeroizeEE** command. Provide a slot number if the encryption engine is a blade.

   ```
   SecurityAdmin:switch> cryptocfg --zeroizeEE
   This will zeroize all critical security parameters
   ARE YOU SURE  (yes, y, no, n): [no]y
   Operation succeeded.
   ```

   Zeroization leaves the switch or blade in the fault state. The switch or blade is rebooted automatically.
3. Synchronize the time on the switch and the key manager appliance. They should be within one minute of each other. Differences in time can invalidate certificates and cause key vault operations to fail.
4. Initialize the node by entering the **cryptocfg --initnode** command. Successful execution generates the following security parameters and certificates:
   - Node CP certificate.
   - Key Archive Client Certificate Signing Request (KAC CSR).

   NOTE
   Node initialization overwrites any existing authentication data on the node.

   ```
   SecurityAdmin:switch> cryptocfg --initnode
   This will overwrite all identification and authentication data
   ARE YOU SURE  (yes, y, no, n): [no] y

   Notify SPM of Node Cfg
   Operation succeeded.
   ```

5. Initialize the encryption engine using the **cryptocfg --initEE** command. Provide a slot number if the encryption engine is a blade. This step generates critical security parameters (CSPs) and certificates in the CryptoModule's security processor (SP). The CP and the SP perform a certificate exchange to register respective authorization data.

   ```
   SecurityAdmin:switch> cryptocfg --initEE
   This will overwrite previously generated identification
   and authentication data
   ARE YOU SURE (yes, y, no, n): y
   Operation succeeded.
   ```

6. Register the encryption engine by entering the **cryptocfg --regEE** command. Provide a slot number if the encryption engine is a blade. This step registers the encryption engine with the CP or chassis. Successful execution results in a certificate exchange between the encryption engine and the CP through the FIPS boundary.

```
SecurityAdmin:switch> cryptocfg --regEE
Operation succeeded.
```

7. Enable the encryption engine by entering the **cryptocfg --enableEE** command.

```
SecurityAdmin:switch> cryptocfg --enableEE
Operation succeeded.
```

8. Repeat the above steps on every node that is expected to perform encryption.

Exporting the KAC certificate signing request (CSR)

You can export the KAC CSR from the switch to file on a LAN-attached host, or you can attach a USB storage device to the switch and export the KAC CSR to that device.

1. Log in to the Brocade Encryption Switch on which the CSR was generated as Admin or SecurityAdmin.
2. Export the CSR from the switch over an SCP-protected LAN connection to a file on an external host (for example, your workstation), or to a mounted USB device.

   The following example exports a CSR to an external SCP-capable host at IP address 192.168.38.245.

```
SecurityAdmin:switch> cryptocfg --export -scp -KACcsr \
192.168.38.245 mylogin /tmp/certs/kac_dpm_cert.pem
Password:
Operation succeeded.
```

   The following example exports a CSR to USB storage.

```
SecurityAdmin:switch> cryptocfg --export -usb KACcsr kac_dpm_cert.pem
Operation succeeded.
```

   If you export the CSR to a USB storage device, you must remove the storage device from the switch and attach it to a computer that has access to a third-party CA. The CSR must be submitted to a CA.

NOTE
The CSR is exported in Privacy Enhanced Mail (.pem) format. The is the format required in exchanges with certificate authorities.

Submitting the CSR to a CA

The CSR must be submitted to a CA to be signed. The CA is a trusted third-party entity that signs the CSR. Several CAs are available and procedures vary, but the general steps are as follows:

1. Open an SSL connection to an X.509 server.
2. Submit the CSR for signing.

3. Request the signed certificate.

   Generally, a public key, the signed KAC certificate, and a signed CA certificate are returned.

4. Download and store the signed certificates.

The following example submits a CSR to the demoCA from RSA.

```
cd /opt/CA/demoCA
openssl x509 -req -sha1 -CAcreateserial -in certs/<Switch CSR Name> -days 365
-CAcacert.pem -CAkey private/cakey.pem -out newcerts/<Switch Cert Name>
```

NOTE
You can change the number of days that a certificate will expire based on your site's security policies. For more information on changing the certificate expiry date, refer to “KAC certificate registration expiry” on page 238.

Importing the signed KAC certificate

The signed KAC certificate must be imported into the Brocade Encryption Switch or blade that generated the CSR and then registered. You can import the signed KAC certificate to the switch from a file on a LAN attached host, or you can write it to a USB storage device, attach the USB storage device to the switch or blade, and import the certificate from that device. The following describes both options:

1. Log in to the switch on which to import the certificate as Admin or SecurityAdmin.
2. Enter the **cryptocfg --import** command with the appropriate parameters.

   The following example imports a certificate named kac_signed_cert.pem that was previously exported to the external host 192.168.38.245. Certificates are imported to a predetermined directory on the node.

```
SecurityAdmin:switch> cryptocfg --import -scp kac_signed_cert.pem \
192.168.38.245 mylogin /tmp/certs/kac_signed_cert.pem
Password:
Operation succeeded.
```

The following example imports a certificate named kac_signed_cert.pem that was previously exported to USB storage.

```
SecurityAdmin:switch> cryptocfg --import -usb kac_signed_cert.pem \
kac_signed_cert.pem
Operation succeeded.
```

3. Register the KAC certificate.

```
SecurityAdmin:switch> cryptocfg --reg -KACcert kac_signed_cert.pem primary
```

4. Repeat steps 1 through 3 for every node in the encryption group.

Uploading the CA certificate onto the DPM appliance (and first-time configurations)

Install the signing authority certificate (CA certificate) on the DPM appliance.

1. Start a web browser and connect to the DPM appliance setup page. You will need the URL, and have the proper authority level, a user name, and a password.
2. Select the **Operations** tab.
3. Select **Certificate Upload**.
4. In the **SSLCAcertificateFile** field, enter the full local path of the CA certificate. Do not use the UNC naming convention format.
5. Select **Upload**, **Configure SSL**, and **Restart Webserver**.
6. After the web server restarts, enter the root password.
7. Open another web browser window, and start the RSA management user interface.

   You will need the URL, and have the proper authority level, a user name, and a password.

   NOTE
   The Identity Group name used in the next step might not exist in a freshly installed DPM. To establish an Identity Group name, click the **Identity Group** tab and create a name. The name **Hardware Retail Group** is used as an example in the following steps.

8. Select the **Key Classes** tab. For each of the following key classes, perform step a through step h to create the class. The key classes must be created only once, regardless of the number of nodes in your encryption group and regardless of the number of encryption groups that will be sharing this DPM.

   kcn.1998-01.com.brocade:DEK_AES_256_XTS

   kcn.1998-01.com.brocade:DEK_AES_256_CCM

   kcn.1998-01.com.brocade:DEK_AES_256_GCM

   kcn.1998-01.com.brocade:DEK_AES_256_ECB

   a. Click **Create**.
   b. Enter the key name string into the **Name** field.
   c. Select **Hardware Retail Group** for **Identity Group**.
   d. Deselect **Activated Keys Have Duration**.
   e. Select **AES** for **Algorithm**.
   f. Select **256** for **Key Size**.
   g. Select the **Mode** for the respective key classes as follows:

      XTS for Key Class "kcn.1998-01.com.brocade:DEK_AES_256_XTS"

      CBC for Key Class "kcn.1998-01.com.brocade:DEK_AES_256_CCM"

      CBC for Key Class "kcn.1998-01.com.brocade:DEK_AES_256_GCM"

      ECB for Key Class "kcn.1998-01.com.brocade:DEK_AES_256_ECB"

   h. Click **Next**.

i. Repeat step a through step h for each key class.

j. Click **Finish**.

Uploading the KAC certificate onto the DPM apliance (manual identity enrollment)

NOTE
The Brocade Encryption Switch will not use the identity auto enrollment feature that is supported with DPM 3.x servers. You must complete the identity enrollment manually to configure the DPM 3.x server with the Brocade Encryption Switch.

You need to install the switch public key certificate (KAC certificate). For each encryption node, create an identity as follows:

1. Select the **Identities** tab.
2. Click **Create**.
3. Enter a label for the node in the **Name** field. This is a user-defined identifier.
4. Select the **Hardware Retail Group** in the **Identity Groups** field.
5. Select the **Operational User** role in the **Authorization** field.
6. Click **Browse** and select the imported certificate <name>_kac_cert.pem> as the **Identity certificate**.
7. Click **Save**.

NOTE
The CA certificate file referenced in the **SSLCAcertificateFile** field (see step 4) must be imported and registered on the switch designated as an encryption group leader. Note this location before proceeding to "Setting heartbeat signaling values" on page 142.

Creating a Brocade encryption group

An encryption group consists of one or more encryption engines. Encryption groups can provide failover/failback capabilities by organizing encryption engines into Data Encryption Key (DEK) clusters. An encryption group has the following properties:

- It is identified by a user-defined name.
- If an encryption group contains more than one node, the group must be managed from a designated group leader.
- If an encryption group consists of one node only, that node must be defined as an encryption group leader.
- All group members must share the same key manager.
- The same master key is used for all encryption operations in the group.
- In the case of FS8-18 blades:
  - All encryption engines in a chassis are part of the same encryption group.
  - An encryption group may contain up to four DCX Backbone nodes with a maximum of four encryption engines per node, forming a total of 16 encryption engines.

To create a Brocade encryption group, complete the following steps:

1. Identify one node (a Brocade Encryption Switch or DCX Backbone chassis with an FS8-18 blade) as the designated group leader and log in as Admin or SecurityAdmin.
2. Enter the **cryptocfg --create -encgroup** command followed by a name of your choice. The name can be up to 15 characters long, and can include any alphanumeric characters and underscores. White space or other special characters are not permitted.

   The following example creates the encryption group **brocade.**

   ```
   SecurityAdmin:switch> cryptocfg --create -encgroup brocade
   Encryption group create status: Operation Succeeded.
   ```

The switch on which you create the encryption group becomes the designated group leader. After you have created an encryption group, all group-wide operations are performed on the group leader.

Client registration for manual enrollment

When you migrate to Fabric OS 7.1.0 from an earlier Fabric OS version, client registration is performed automatically and no user intervention is required during the upgrade process. For new deployments, however, identity enrollment must be performed manually for the Brocade Encryption Switch to connect with the DPM 3.x servers. Refer to "Steps for connecting to a DPM appliance" on page 134.

Once completed, client registration occurs after key vault registration, when the Brocade Encryption Switch attempts to connect to the DPM server for the first time.

During registration of the key vault on the Brocade Encryption Switch, the following configuration files are created:

- Init file: This file is created under /etc/fabos/certs/sw0/DpmInit.cfg. The init file contains static configuration information. A sample init file is provided.

  ```
  svcType=transportSvc
  configName=https_cfg_1
  clientCredentialFile=/etc/fabos/certs/sw0/kac.p12
  clientTrustedRoots=/etc/fabos/certs/sw0/kv.pem
  clientCredentialPassword=Password1
  client.registrationfile=/etc/fabos/certs/sw0/DpmReg_10.37.39.33
  address=https://10.37.39.33/KMS/rpc/emu
  port=443
  responseTimeout=10
  connectRetries=0
  connectTimeout=10
  certHostnameVerification=false
  FIPSMode=false
  svcType=cacheSvc
  configName=cache_cfg_1
  nonPersistentCache=false
  persistentCacheFile=
  applicationId=B10_00_00_05_1e_53_89_eb
  cachePassword=Password1
  svcType=logSvc
  configName=log_cfg_1
  error=false
  warning=false
  audit=false
  ```

- Registration File: This file is created as /etc/fabos/certs/sw0/DpmReg_<KV IP>. The registration file contains the current registration status of the client. A sample registration file before successful registration with the DPM server is provided.

  ```
  client.registration_state = 0
  client.actmgmt_enable = 0
  client.app_name = B10_00_00_05_1e_53_89_eb
  client.actmgmt_poll_interval = 0
  ```

  During a successful key vault connection and client registration, the registration file is updated with the information provided by the DPM server. A sample registration file showing successful registration with the DPM server is provided.

  ```
  client.actmgmt_enable = 0
  client.policy_signature =
  jGvUbFqw4iw64YB0MHrSbMeaVN9hd7EXFFQkUFMxd71kUd0NWSjl/pJO5mt4quppYdsvllgyXg
  e8NdTbvsOGdtDGJxjpvRjQyi4YqWm/kzLiFlwRGMKcf2kkhDMdW3Is+cvUBmLJNiPkNCJ7xAYV
  aJ2tpWiQ/mBJfrTw7uuCcZY=
  client.rkm_svr_public_key =
  MIGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKBgQC1Q73enodAh4FOY3YobU7d+DO6LZynnzbmYQ
  Sztn+JzxuuvUgzakKtKJO5FD+nWnDpcz9dd8ZHY7Gq2IMQBl5GL8Sjw6eGOklw7qG5Lojlcuiz
  XJ6hBk7sp1fEw1PRHb5v219IqoplAVB8masw+eYb9T0gssQQRepTGrmqzCCAXQIDAQAB
  client.app_id =
  0eeef136-f84b-4cd8-9d1a-1c5cdf86bd12-db26194e-9883-4aa6-8d62-ca98420fd016
  client.policy_name = DEFAULT_POLICY
  client.applicationpolicy = 000102030405060708091011
  client.app_name = B10_00_00_05_1e_53_89_eb
  client.registration_state = 3
  ```

  When the registration process is completed successfully, the registration state will be 3. If the state is 0, registration is in progress.

DPM key vault high availability deployment

When dual DPM appliances are used for high availability, the DPM appliances must be clustered and must operate in maximum availability mode, as described in the DPM appliance user documentation.

When dual DPM appliances are clustered, they are accessed using an IP load balancer. For a complete high-availability deployment, the multiple IP load balancers are clustered, and the IP load balancer cluster exposes a virtual IP address called a floating IP address. The floating IP address must be registered on the Fabric OS encryption group leader using the **cryptocfg** **--reg** **-keyvault** command.

Neither the secondary DPM appliance nor individual DPM appliance IP addresses should be registered. The command to register a secondary DPM appliance is blocked, beginning with Fabric OS 6.3.0.

Setting heartbeat signaling values

Encryption group nodes use heartbeat signaling to communicate to one another and to their associated key vaults. The default heartbeat signaling values are three retries (heartbeat misses) with a two-second timeout (heartbeat timeout) between each retry. If three consecutive heartbeats are missed (the equivalent of six seconds without contact), the encryption group leader node declares a member node as unreachable, resulting in an encryption group split scenario (EG split).

It is highly recommended that all nodes comprising your encryption group and your key vaults be a part of a dedicated management LAN or on a LAN that is stable and not congested to avoid the possibility of an EG split. The default values are appropriate for a LAN that is stable and not congested.

In the unlikely scenario of an EG split, the encryption group automatically begins an auto-recovery process. No user intervention is required unless the congestion in the network or network loss is prolonged or continuous. Under such conditions, auto-recovery will most likely fail, as the encryption group leader node will not be able to establish a clean series of heartbeats with the other member nodes. Refer to "EG split possibilities requiring manual recovery" on page 258 for manual recovery procedures.

If the management network becomes congested or unreliable, resulting in excessive auto-recovery processing or the need for manual recovery from EG splits, it is possible to set larger heartbeat and heartbeat timeout values to mitigate the chances of having the EG split while the network issues are being addressed. The following commands are issued from the encryption group leader nodes to change the heartbeat signaling values.

```
switch:admin-> cryptocfg -set -hbmisses <number>
switch:admin-> cryptocfg -set -hbtimeout <time>
```

Where:

<number> Sets the number of heartbeat misses allowed in a node that is part of an encryption group before the node is declared unreachable. This value is set in conjunction with the timeout value. It must be configured at the group leader node and is distributed to all member nodes in the encryption group. The value entered specifies the number of heartbeat misses. The default value is 3. Valid values are integers ranging from 3–14.

<time> Sets the timeout value for the heartbeat. This parameter must be configured at the group leader node and is distributed to all member nodes in the encryption group. The value entered specifies the heartbeat timeout in seconds. The default value is 2 seconds. Valid values are integers ranging from 2–9.

NOTE
The collective time allowed (the heartbeat timeout value multiplied by the heartbeat misses) cannot exceed 30 seconds. (This is enforced by Fabric OS.)

If the group leader is the only member in the encryption group, proceed to "Registering DPM on a Fabric OS encryption group leader" on page 145.

To add encryption group members, see "Adding a member node to an encryption group" on page 143.

Adding a member node to an encryption group

During the initialization phase, a set of key pairs and certificate is generated on every node. The certificates are used for mutual identification and authentication with other group members and with DPM. Every device must have a certificate to participate in the deployment of encryption services. Some devices must have each other's certificates in order to communicate.

Before adding a member node to an encryption group, ensure that the node has been properly initialized and that all encryption engines are in an enabled state. See "Initializing the Fabric OS encryption engines" on page 135.

After adding a member node to the encryption group, the following operations can still be performed on the member node, if necessary. Initially, these commands should not be necessary if the initialization procedure was followed:

- cryptocfg --initEE
- cryptocfg --regEE
- cryptocfg --enableEE

CAUTION

After adding the member node to the encryption group, you should not use the cryptocfg --zeroizeEE command on that node. Doing so removes critical information and makes it necessary to reinitialize the node and export the new CP certificates and KAC certificates to the group leader and the key vault.

To add a member node to an encryption group, complete the following steps:

1. Log in to the switch on which the certificate was generated as Admin or FabricAdmin.
2. Execute the **cryptocfg --reclaimWWN -cleanup** command.
3. Log in as Admin or SecurityAdmin.
4. Export the certificate from the local switch to an SCP-capable external host or to a mounted USB device. Enter the **cryptocfg --export** command with the appropriate parameters. When exporting a certificate to a location other than your home directory, you must specify a fully qualified path that includes the target directory and file name. When exporting to USB storage, certificates are stored by default in a predetermined directory, and you only need to provide a file name for the certificate. The file name must be given a **.pem** (privacy enhanced mail) extension. Use a character string that identifies the certificate's originator, such as the switch name or IP address.

   The following example exports a CP certificate from an encryption group member to an external SCP-capable host and stores it as enc_switch1_cp_cert.pem.

   ```
   SecurityAdmin:switch> cryptocfg --export -scp CPcert \
   192.168.38.245 mylogin /tmp/certs/enc_switch1_cp_cert.pem
   Password:
   Operation succeeded.
   ```

   The following example exports a CP certificate from the local node to USB storage.

   ```
   SecurityAdmin:switch>cryptocfg --export -usb CPcert enc_switch1_cp_cert.pem
   Operation succeeded.
   ```

5. Use the cryptocfg --import command to import the CP certificates to the group leader node. You must import the CP certificate of each node you wish to add to the encryption group.

   The following example imports a CP certificate named “enc_switch1_cp_cert.pem” that was previously exported to the external host 192.168.38.245. Certificates are imported to a predetermined directory on the group leader.

   ```
   SecurityAdmin:switch> cryptocfg --import -scp enc_switch1_cp_cert.pem \
   192.168.38.245 mylogin /tmp/certs/enc_switch1_cp_cert.pem
   Password:
   Operation succeeded.
   ```

   The following example imports a CP certificate named “enc_switch1_cp_cert.pem” that was previously exported to USB storage:

   ```
   SecurityAdmin:switch> cryptocfg --import -usb enc_switch1_cp_cert.pem \
   enc_switch1_cp_cert.pem
   Operation succeeded.
   ```

NOTE
If the maximum number of certificates is exceeded, the following message is displayed.

```
Maximum number of certificates exceeded.  Delete an unused certificate with the
‘cryptocfg –delete –file’ command and then try again.
```

6. Enter the cryptocfg --show -file -all command on the group leader to verify that you have imported all necessary certificates.

   The following example shows the member node CP certificate that was imported earlier to the group leader.

   ```
   SecurityAdmin:switch> cryptocfg --show -file -all
   File name: enc_switch1_cp_cert.pem, size: 1338 bytes
   ```

7. On the group leader, register each node you are planning to include in the encryption group. Enter the cryptocfg --reg -membernode command with appropriate parameters to register the member node. Specify the member node’s WWN, Certificate filename, and IP address when executing this command. Successful execution of this command distributes all necessary node authentication data to the other members of the group.

   ```
   SecurityAdmin:switch> cryptocfg --reg -membernode \
   10:00:00:05:1e:39:14:00 enc_switch1_cert.pem 10.32.244.60
   Operation succeeded.
   ```

   NOTE
   The order in which member node registration is performed defines group leader succession. At any given time there is only one active group leader in an encryption group. The group leader succession list specifies the order in which group leadership is assumed if the current group leader is not available.

8. Display encryption group member information. This example shows the encryption group brocade with two member nodes, one group leader and one regular member. No key vault or HA cluster is configured, and the values for master key IDs are zero.

   ```
   SecurityAdmin:switch> cryptocfg --show -groupmember -all
   NODE LIST
   Total Number of defined nodes:2
   Group Leader Node Name:     10:00:00:05:1e:41:9a:7e
   ```

```
Encryption Group state:     CLUSTER_STATE_CONVERGED

Node Name:                  10:00:00:05:1e:41:9a:7e (current node)
 State:                     DEF_NODE_STATE_DISCOVERED
 Role:                      GroupLeader
 IP Address:                10.32.244.71
 Certificate:               GL_cpcert.pem
 Current Master Key State:  Not configured
 Current Master KeyID:      00:00:00:00:00:00:00:00:00:00:00:00:00:00:00:00
 Alternate Master Key State:Not configured
 Alternate Master KeyID:    00:00:00:00:00:00:00:00:00:00:00:00:00:00:00:00

 EE Slot: 0
  SP state:                 Operational; Need Valid KEK
  Current Master KeyID:     00:00:00:00:00:00:00:00:00:00:00:00:00:00:00:00
  Alternate Master KeyID:   00:00:00:00:00:00:00:00:00:00:00:00:00:00:00:00
  No HA cluster membership

Node Name:                  10:00:00:05:1e:39:14:00
 State:                     DEF_NODE_STATE_DISCOVERED
 Role:                      MemberNode
 IP Address:                10.32.244.60
 Certificate:               enc1_cpcert.pem
 Current Master Key State:  Not configured
 Current Master KeyID:      00:00:00:00:00:00:00:00:00:00:00:00:00:00:00:00
 Alternate Master Key State:Not configured
 Alternate Master KeyID:    00:00:00:00:00:00:00:00:00:00:00:00:00:00:00:00

 EE Slot:     0
  SP state:                 Unknown State
  Current Master KeyID:     00:00:00:00:00:00:00:00:00:00:00:00:00:00:00:00
  Alternate Master KeyID:   00:00:00:00:00:00:00:00:00:00:00:00:00:00:00:00
  No HA cluster membership
```

Registering DPM on a Fabric OS encryption group leader

You will need to know the download location for the CA certificate. The path to the file was entered in the **SSLCAcertificateFile** field when "Uploading the CA certificate onto the DPM appliance (and first-time configurations)" on page 138. Also, if you are using an DPM cluster for high availability, you will need the virtual IP address, as described in "DPM key vault high availability deployment" on page 141.

1. Log in as Admin or SecurityAdmin.
2. Set the key vault type to DPM by entering the **cryptocfg --set -keyvault** command. Successful execution sets the key vault type for the entire encryption group. The following example sets the keyvault type to DPM:

   ```
   SecurityAdmin:switch> cryptocfg --set -keyvault DPM
   Set key vault status: Operation Succeeded.
   ```

3. Import and register DPM on the group leader using the CA certificate for the CA that signed the DPM key vault certificate. The group leader automatically shares this information with other group members. It might take a minute to complete the operation.

   ```
   SecurityAdmin:switch> cryptocfg --import -scp <CA certificate file>
   <host IP> <host username> <host path>
   ```

```
SecurityAdmin:switch> cryptocfg --reg -keyvault <CA certificate file>
<DPM IP> primary
```

NOTE
If you are using an DPM cluster for high availability, the IP address specified as `<DPM IP>` is the virtual IP address of the DPM cluster.

4. As the switches come up, enable the encryption engines.

```
SecurityAdmin:switch> cryptocfg --enableEE
Operation succeeded.
```

Generating and backing up the master key

You must generate a master key on the group leader, and export it to a secure backup location so that it can be restored, if necessary. The master key is used to encrypt DEKs for transmission to and from a DPM.

The backup location may be a DPM, a local file, or a secure external SCP-capable host. All three options are shown in the following procedure. Note that the Brocade SAN Management application provides the additional option of backing up the master key to system cards.

1. Generate the master key on the group leader.

```
SecurityAdmin:switch> cryptocfg --genmasterkey
Master key generated. The master key should be
exported before further operations are performed.
```

2. Export the master key to the key vault. Make a note of the key ID and the passphrase. You will need the Key ID and passphrase should you have to restore the master key from the key vault.

```
SecurityAdmin:switch> cryptocfg --exportmasterkey
Enter the passphrase: passphrase
Master key exported. Key ID:
8f:88:45:32:8e:bf:eb:44:c4:bc:aa:2a:c1:69:94:2
```

3. Save the master key to a file.

```
SecurityAdmin:switch> cryptocfg --exportmasterkey -file
Master key file generated.
```

4. Export the master key to an SCP-capable external host:

```
SecurityAdmin:switch> cryptocfg --export -scp -currentMK \
192.168.38.245 mylogin GL_MK.mk
Password:
Operation succeeded.
```

5. Display the group configuration.

```
SecurityAdmin:switch> cryptocfg --show -groupcfg
Encryption Group Name:       brocade
  Failback mode:             Manual
  Heartbeat misses:          3
  Heartbeat timeout:         2
```

```
      Key Vault Type:            DPM
    Primary Key Vault:
      IP address:                10.33.54.160
      Certificate ID:            HPDPM_CA1
      Certificate label:        DPMCERT
      State:                     Connected
      Type:                      DPM
    Secondary Key Vault not configured
    NODE LIST
    Total Number of defined nodes:  2
    Group Leader Node Name:         10:00:00:05:1e:41:9a:7e
    Encryption Group state:         CLUSTER_STATE_CONVERGED
           Node Name            IP address        Role
    10:00:00:05:1e:41:9a:7e   10.32.244.71   GroupLeader(current node)
    10:00:00:05:1e:39:14:00   10.32.244.60   MemberNode
```

6. Display the group membership information. Verify that the master key ID for all member nodes is the same.

```
SecurityAdmin:switch> cryptocfg --show -groupmember -all
NODE LIST
Total Number of defined nodes:2
Group Leader Node Name:    10:00:00:05:1e:41:9a:7e
Encryption Group state:    CLUSTER_STATE_CONVERGED

Node Name:                  10:00:00:05:1e:41:9a:7e (current node)
 State:                     DEF_NODE_STATE_DISCOVERED
 Role:                      GroupLeader
 IP Address:                10.32.244.71
 Certificate:               GL_cpcert.pem
 Current Master Key State:  Configured
 Current Master KeyID:      8f:88:45:32:8e:bf:eb:44:c4:bc:aa:2a:c1:69:94:2
 Alternate Master Key State: Not configured
 Alternate Master KeyID:    00:00:00:00:00:00:00:00:00:00:00:00:00:00:00:00

 EE Slot:                   0
  SP state:                 Waiting for enableEE
  Current Master KeyID:     8f:88:45:32:8e:bf:eb:44:c4:bc:aa:2a:c1:69:94:2
  Alternate Master KeyID:   00:00:00:00:00:00:00:00:00:00:00:00:00:00:00:00
  No HA cluster membership

Node Name:                  10:00:00:05:1e:39:14:00
 State:                     DEF_NODE_STATE_DISCOVERED
 Role:                      MemberNode
 IP Address:                10.32.244.60
 Certificate:               enc1_cpcert.pem
 Current Master Key State:  Configured
 Current Master KeyID:      8f:88:45:32:8e:bf:eb:44:c4:bc:aa:2a:c1:69:94:2
 Alternate Master Key State: Not configured
 Alternate Master KeyID:    00:00:00:00:00:00:00:00:00:00:00:00:00:00:00:00

 EE Slot:     0
  SP state:                 Waiting for enableEE
  Current Master KeyID:     8f:88:45:32:8e:bf:eb:44:c4:bc:aa:2a:c1:69:94:2
  Alternate Master KeyID:   00:00:00:00:00:00:00:00:00:00:00:00:00:00:00:00
  No HA cluster membership
```

7. Display encryption group member information. This example shows the encryption group brocade with two member nodes, one group leader and one regular member. No key vault or HA cluster is configured, and the values for master key IDs are zero.

```
SecurityAdmin:switch> cryptocfg --show -groupmember -all
NODE LIST
Total Number of defined nodes:2
Group Leader Node Name:     10:00:00:05:1e:41:9a:7e
Encryption Group state:     CLUSTER_STATE_CONVERGED

Node Name:                  10:00:00:05:1e:41:9a:7e (current node)
 State:                     DEF_NODE_STATE_DISCOVERED
 Role:                      GroupLeader
 IP Address:                10.32.244.71
 Certificate:               GL_cpcert.pem
 Current Master Key State:  Not configured
 Current Master KeyID:      00:00:00:00:00:00:00:00:00:00:00:00:00:00:00:00
 Alternate Master Key State:Not configured
 Alternate Master KeyID:    00:00:00:00:00:00:00:00:00:00:00:00:00:00:00:00

 EE Slot: 0
  SP state:                 Operational; Need Valid KEK
  Current Master KeyID:     00:00:00:00:00:00:00:00:00:00:00:00:00:00:00:00
  Alternate Master KeyID:   00:00:00:00:00:00:00:00:00:00:00:00:00:00:00:00
  No HA cluster membership

Node Name:                  10:00:00:05:1e:39:14:00
 State:                     DEF_NODE_STATE_DISCOVERED
 Role:                      MemberNode
 IP Address:                10.32.244.60
 Certificate:               enc1_cpcert.pem
 Current Master Key State:  Not configured
 Current Master KeyID:      00:00:00:00:00:00:00:00:00:00:00:00:00:00:00:00
 Alternate Master Key State:Not configured
 Alternate Master KeyID:    00:00:00:00:00:00:00:00:00:00:00:00:00:00:00:00

 EE Slot:     0
  SP state:                 Unknown State
  Current Master KeyID:     00:00:00:00:00:00:00:00:00:00:00:00:00:00:00:00
  Alternate Master KeyID:   00:00:00:00:00:00:00:00:00:00:00:00:00:00:00:00
  No HA cluster membership
```

High availability clusters

A high availability (HA) cluster consists of exactly two encryption engines configured to host the same CryptoTargets and to provide Active/Standby failover and failback capabilities in a single fabric. Failback occurs automatically by default, but is configurable with a manual failback option. All encryption engines in an encryption group share the same DEK for a disk or tape LUN.

HA cluster configuration rules

The following rules apply when configuring an HA cluster:

- The encryption engines that are part of an HA cluster must belong to the same encryption group and be part of the same fabric.
- An HA cluster cannot span fabrics and it cannot provide failover/failback capability within a fabric transparent to host MPIO software.
- HA cluster configuration and related operations must be performed on the group leader.

- HA clusters of FS8-18 blades should not include blades in the same DCX Backbone chassis.

NOTE
In Fabric OS 6.3.0 and later, HA cluster creation is blocked when encryption engines belonging to FS8-18 blades in the same DCX Backbone Chassis are specified.

- Cluster links must be configured before creating an HA cluster. Refer to the section "Configuring cluster links" on page 131 for instructions.
- Configuration changes must be committed before they take effect. Any operation related to an HA cluster that is performed without a commit operation will not survive across switch reboots, power cycles, CP failover, or HA reboots.
- It is recommended that the HA cluster configuration be completed before you configure storage devices for encryption.
- It is mandatory that the two encryption engines in the HA cluster belong to two different nodes for true redundancy. This is always the case for Brocade Encryption Switches, but is not true if two FS8-18 blades in the same DCX Backbone Chassis are configured in the same HA cluster. In Fabric OS v6.3.0 and later releases, HA cluster creation is blocked when encryption engines belonging to FS8-18 blades in the same DCX Backbone Chassis are specified.

Creating an HA cluster

1. Log in to the group leader as Admin or FabricAdmin.
2. Enter the **cryptocfg --create -hacluster** command. Specify a name for the HA cluster and optionally add the node WWN of the encryption engine you wish to include in the HA cluster. Provide a slot number if the encryption engine is a blade. The following example creates an HA cluster named "HAC1" with two encryption engines.

```
FabricAdmin:switch> cryptocfg --create -hacluster HAC 10:00:00:05:1
e:51:94:00 2 10:00:00:05:1e:55:3a:f0 0
                    Slot     Local/
      EE Node WWN  Number    Remote
10:00:00:05:1e:51:94:00     2    Local
                    Slot     Local/
      EE Node WWN  Number    Remote
10:00:00:05:1e:55:3a:f0     0    Remote
Operation succeeded.
```

3. Enter **cryptocfg --commit** to commit the transaction. Any transaction remains in the **defined** state until it is committed. The commit operation fails if the HA cluster has less than two members.
4. Display the HA cluster configuration by entering the **cryptocfg --show -hacluster -all** command. In the following example, the encryption group **brocade** has one committed HAC1 with two encryption engines.

```
FabricAdmin:switch> cryptocfg --show -hacluster -all
Encryption Group Name: brocade
Number of HA Clusters: 1

HA cluster name: HAC1 - 1 EE entry
Status:          Committed
         WWN             Slot Number  Status
11:22:33:44:55:66:77:00      0         Online
10:00:00:05:1e:53:74:87      3         Online
```

NOTE
An HA cluster configuration must have two encryption engines before you can commit the transaction with the **cryptocfg --commit** command. To commit an incomplete HA cluster, you have the option to force the commit operation by issuing **cryptocfg --commit -force.** Use the forced commit with caution, because the resulting configuration will not be functional and provide no failover/failback capabilities.

Adding an encryption engine to an HA cluster

1. Log in to the group leader as Admin or FabricAdmin.
2. Enter the **cryptocfg --add -haclustemember** command. Specify the HA cluster name and the encryption engine node WWN. Provide a slot number if the encryption engine is a blade. The following example adds a Brocade FS8-18 in slot 5 to the HA cluster HAC2.

```
FabricAdmin:switch>cryptocfg --add -haclustermember HAC2 \
10:00:00:60:5b:03:1c:90 5
EE Node WWN: 10:00:00:60:5b:03:1c:90 5 Slot number: 5Detected
Add HA cluster member status: Operation succeeded.
```

3. Add another encryption engine before committing the transaction.

NOTE
You cannot add the same node to the HA cluster.

Removing engines from an HA cluster

Removing the last engine from an HA cluster also removes the HA cluster. If only one engine is removed from the cluster, you must either add another engine to the cluster, or remove the other engine.

1. Log in to the group leader as Admin or FabricAdmin.
2. Enter the **cryptocfg --remove -haclustemember** command. Specify the HA cluster name and the encryption engine node WWN. Provide a slot number if the encryption engine is a blade. The following example removes a Brocade FS8-18 in slot 5 from the HA cluster HAC2.

```
FabricAdmin:switch> cryptocfg --remove -haclustermember HAC2 \
10:00:00:60:5b:03:1c:90 5
EE Node WWN: 10:00:00:60:5b:03:1c:90 5 Slot number: 5Detected
Remove HA cluster member status: Operation succeeded.
```

3. Either remove the second engine or add a replacement second engine, making sure all HA clusters have exactly two engines.

Swapping engines in an HA cluster

Swapping engines is useful when replacing hardware. Swapping engines is different from removing an engine and adding another because when you swap engines, the configured targets on the former HA cluster member are moved to the new HA cluster member.

1. Log in to the group leader as Admin or FabricAdmin.
2. Enter the **cryptocfg --replace** [-**haclustermember** <HA cluster name>] command

```
   <<old node WWN> [old slot number]>
   <<new node WWN> [new slot number]>:

HA cluster name: dthac - 2 EE entries
Status:          Committed
HAC State:       Converged

          WWN             Slot Number   Status
10:00:00:05:1e:39:a6:7e        4        Online
10:00:00:05:1e:c1:06:63        0        Online

sw153114:FID128:admin> cryptocfg --replace -haclustermember dthac
10:00:00:05:1e:39:a6:7e 4 10:00:00:05:1e:39:a6:7e 12
                           Slot      Local/
      EE Node WWN         Number    Remote
10:00:00:05:1e:39:a6:7e     12      Local
Operation succeeded.
```

After:

```
sw153114:FID128:admin> cryptocfg --show -hacluster -all
Encryption Group Name: disk_tape
Number of HA Clusters: 1

HA cluster name: dthac - 2 EE entries
Status:          Defined
HAC State:       Converged

          WWN             Slot Number   Status
10:00:00:05:1e:39:a6:7e       12        Online
10:00:00:05:1e:c1:06:63        0        Online
sw153114:FID128:admin>
```

NOTE
The two engines being swapped must be in the same fabric.

Failover/failback policy configuration

Failover/failback policy parameters as outlined in Table 5 can be set for the entire encryption group on the group leader.

Use the **cryptocfg --set** command with the appropriate parameter to set the values for the policy. Policies are automatically propagated to all member nodes in the encryption group.

TABLE 5 Group-wide policies

Policy name	cryptocfg --set parameters	Description
Failover policy	–failbackmode auto | manual	Sets the failback mode. Valid values for failback mode are: • auto - Enables automatic failback mode. Failback occurs automatically within an HA cluster when an encryption switch or blade that failed earlier has been restored or replaced. Automatic failback mode is enabled by default. • manual - Enables manual failback mode. In this mode, failback must be initiated manually when an encryption switch or blade that failed earlier has been restored or replaced.
Heartbeat misses	–hbmisses *value*	Sets the number of Heartbeat misses allowed in a node that is part of an encryption group before the node is declared unreachable and the standby takes over. The default *value* is 3. The range is 3-14 in integer increments only.
Heartbeat timeout	–hbtimeout *value*	Sets the time-out value for the Heartbeat in seconds. The default *value* is 2 seconds. Valid *values* are integers in the range between 2 and 9 seconds. NOTE: The relationship between –hbmisses and –hbtimeout determines the total amount of time allowed before a node is declared unreachable. If a switch does not sense a heartbeat within the heartbeat timeout value, it is counted as a heartbeat miss. The default values result in a total time of 6 seconds (timeout value of two seconds times three misses). A total time of 6–9 seconds is recommended. A smaller value may cause a node to be declared unreachable prematurely, while a larger value could result in inefficiency.

Policy Configuration Examples

The following examples illustrate the setting of group-wide policy parameters.

To set the failback mode to manual failback:

```
SecurityAdmin:switch> cryptocfg --set -failbackmode manual
Set failback policy status: Operation Succeeded.
```

To set the Heartbeat misses value to 3:

```
SecurityAdmin:switch> cryptocfg --set -hbmisses 3
Set heartbeat miss status: Operation Succeeded.
```

To set the Heartbeat timeout value to 3 seconds:

```
SecurityAdmin:switch> cryptocfg --set -hbtimeout 3
Set heartbeat timeout status: Operation Succeeded.
```

Re-exporting a master key

With the introduction of Fabric OS v7.0.0, you can export master keys to the key vault multiple times instead of only once. The ability to export the master key more than once enables you to recover the master key when needed. For example, prior to Fabric OS 7.0.0, if you forgot your passphrase that was used to export the master key, you were not able to recover the master key from the key vault. The ability to re-export the master key in this scenario alleviates this concern.

When the master key is exported to the key vault for the first time, it is stored with the actual master key ID. Subsequent exports are provided with additional exported key IDs that are generated by the Brocade Encryption Switch. Each additional time the master key is exported to the key vault, a different key ID is saved.

The master key can be recovered from any export using the exported master key ID and the corresponding passphrase.

Note the following:

- If you are upgrading to Fabric OS v7.0.0 from an earlier version, (for example, Fabric OS v6.4.x), you can recover the master key with the master key ID. Additional exports of the master key are allowed with the exported master key IDs.
- If you are downgrading from Fabric OS v7.0.0 to an earlier version (for example, Fabric OS v6.4.x), you can recover the master key using the master key ID that is exported in Fabric OS v7.0.0 and its corresponding passphrase. Downgrading to earlier versions allows the master key to be recoverable with only the master key ID.

The --**show** -**localEE** command shows the actual master key IDs, along with the new master key IDs. Also shown are all exported master key IDs associated with a given (actual) master key.

NOTE
You will need to remember the exported master key ID and passphrase you used while exporting the master key ID.

A new subcommand is available to support exporting master key IDs for a given master key.

```
SecurityAdmin:switch> cryptocfg --show -mkexported_keyids <MK ID>
```

The following example lists the exported master key IDs for a given master key ID:

```
SecurityAdmin:switch> cryptocfg --show -mkexported_keyids
e3:ae:aa:89:ec:12:0c:04:29:61:9c:99:44:a3:9b:92

e3:ae:aa:89:ec:12:0c:04:29:61:9c:99:44:a3:9b:92
e3:ae:aa:89:ec:12:0c:04:29:61:9c:99:44:a3:9b:93
e3:ae:aa:89:ec:12:0c:04:29:61:9c:99:44:a3:9b:94
e3:ae:aa:89:ec:12:0c:04:29:61:9c:99:44:a3:9b:95
e3:ae:aa:89:ec:12:0c:04:29:61:9c:99:44:a3:9b:96
e3:ae:aa:89:ec:12:0c:04:29:61:9c:99:44:a3:9b:97
e3:ae:aa:89:ec:12:0c:04:29:61:9c:99:44:a3:9b:98
e3:ae:aa:89:ec:12:0c:04:29:61:9c:99:44:a3:9b:99
e3:ae:aa:89:ec:12:0c:04:29:61:9c:99:44:a3:9b:9a
e3:ae:aa:89:ec:12:0c:04:29:61:9c:99:44:a3:9b:9b
Operation succeeded.
```

The exported key ID is displayed with the master key ID, as shown in the examples to follow:

Example: Initial master key export

```
SecurityAdmin:switch> cryptocfg --exportmasterkey
```

```
Enter passphrase:

Confirm passphrase:

Master key exported.
MasterKey ID: 1a:e6:e4:26:6b:f3:81:f7:d8:eb:cc:0f:09:7a:a4:7e
Exported Key ID: 1a:e6:e4:26:6b:f3:81:f7:d8:eb:cc:0f:09:7a:a4:7e
```

Exporting an additional key ID

Example: Subsequent master key exports.

```
SecurityAdmin:switch> cryptocfg --exportmasterkey
Enter passphrase:

Confirm passphrase:

Master key exported.
MasterKey ID: 1a:e6:e4:26:6b:f3:81:f7:d8:eb:cc:0f:09:7a:a4:7e
Exported Key ID: 1a:e6:e4:26:6b:f3:81:f7:d8:eb:cc:0f:09:7a:a4:7f

SecurityAdmin:switch> cryptocfg --exportmasterkey
Enter passphrase:

Confirm passphrase:

Master key exported.
MasterKey ID: 1a:e6:e4:26:6b:f3:81:f7:d8:eb:cc:0f:09:7a:a4:7e
Exported Key ID: 1a:e6:e4:26:6b:f3:81:f7:d8:eb:cc:0f:09:7a:a4:80
```

Example: Recovering a master key using master key ID from the second master key export

```
SecurityAdmin:switch> cryptocfg --recovermasterkey currentMK -keyID
15:30:f0:f3:5c:2b:28:ce:cc:a7:b4:cd:7d:2a:91:fc

Enter passphrase:
Recover master key status: Operation Succeeded.
```

Viewing the master key IDs

The `show localEE` command shows the actual master key IDs, along with the new master key IDs. Also shown are all exported master key IDs associated with a given (actual) master key.

NOTE
You will need to remember the exported master key ID and passphrase you used while exporting the master key ID.

A new subcommand is available to support exporting master key IDs for a given master key.

```
SecurityAdmin:switch> cryptocfg --show -mkexported_keyids <MK ID>
```

The following example lists the exported master key IDs for a given master key ID:

```
SecurityAdmin:switch> cryptocfg --show –mkexported_keyids
e3:ae:aa:89:ec:12:0c:04:29:61:9c:99:44:a3:9b:92

e3:ae:aa:89:ec:12:0c:04:29:61:9c:99:44:a3:9b:92
```

```
e3:ae:aa:89:ec:12:0c:04:29:61:9c:99:44:a3:9b:93
e3:ae:aa:89:ec:12:0c:04:29:61:9c:99:44:a3:9b:94
e3:ae:aa:89:ec:12:0c:04:29:61:9c:99:44:a3:9b:95
e3:ae:aa:89:ec:12:0c:04:29:61:9c:99:44:a3:9b:96
e3:ae:aa:89:ec:12:0c:04:29:61:9c:99:44:a3:9b:97
e3:ae:aa:89:ec:12:0c:04:29:61:9c:99:44:a3:9b:98
e3:ae:aa:89:ec:12:0c:04:29:61:9c:99:44:a3:9b:99
e3:ae:aa:89:ec:12:0c:04:29:61:9c:99:44:a3:9b:9a
e3:ae:aa:89:ec:12:0c:04:29:61:9c:99:44:a3:9b:9b
Operation succeeded.
```

The exported key ID is displayed with the master key ID, as shown in the examples to follow:

Example: Initial master key export

```
SecurityAdmin:switch> cryptocfg --exportmasterkey
Enter passphrase:

Confirm passphrase:

Master key exported.
MasterKey ID: 1a:e6:e4:26:6b:f3:81:f7:d8:eb:cc:0f:09:7a:a4:7e
Exported Key ID: 1a:e6:e4:26:6b:f3:81:f7:d8:eb:cc:0f:09:7a:a4:7e
```

Example: Subsequent master key exports

```
SecurityAdmin:switch> cryptocfg --exportmasterkey
Enter passphrase:

Confirm passphrase:

Master key exported.
MasterKey ID: 1a:e6:e4:26:6b:f3:81:f7:d8:eb:cc:0f:09:7a:a4:7e
Exported Key ID: 1a:e6:e4:26:6b:f3:81:f7:d8:eb:cc:0f:09:7a:a4:7f

SecurityAdmin:switch> cryptocfg --exportmasterkey
Enter passphrase:

Confirm passphrase:

Master key exported.
MasterKey ID: 1a:e6:e4:26:6b:f3:81:f7:d8:eb:cc:0f:09:7a:a4:7e
Exported Key ID: 1a:e6:e4:26:6b:f3:81:f7:d8:eb:cc:0f:09:7a:a4:80
```

Example: Recovering a master key using master key ID from the second master key export

```
SecurityAdmin:switch> cryptocfg --recovermasterkey currentMK -keyID
15:30:f0:f3:5c:2b:28:ce:cc:a7:b4:cd:7d:2a:91:fc

Enter passphrase:
Recover master key status: Operation Succeeded.
```

Enabling the encryption engine

Enable the encryption engine by entering the **cryptocfg --enableEE** command. Provide a slot number if the encryption engine is a blade.

NOTE
Every time a Brocade Encryption Switch or DCX Backbone chassis containing one or more FS8-18 blades goes through a power cycle event, or after issuing **slotpoweroff <slot number>** followed by **slotpoweron <slot number>** for an FS8-18 blade in the DCX Backbone chassis, the encryption engine must be enabled manually by the Security Administrator. Hosts cannot access the storage LUNs through the storage paths exposed on this Brocade Encryption Switch or FS8-18 blade until the encryption engine is enabled. The encryption engine state can viewed using the **cryptocfg --show -localEE** command, or by displaying switch or blade properties from DFCM. An encryption engine that is not enabled indicates **Waiting for Enable EE**.

```
SecurityAdmin:switch> cryptocfg --enableEE
Operation succeeded.
```

Checking encryption engine status

You can verify the encryption engine status at any point in the setup process and get information about the next required configuration steps or to troubleshoot an encryption engine that behaves in unexpected ways. Use the **cryptocfg --show -localEE** command to check the encryption engine status.

```
SecurityAdmin:switch> cryptocfg --show -localEE

    EE Slot:                    0
        SP state:               Waiting for initEE
        EE key status not available: SP TLS connection is not up.
        No HA cluster membership
EE Slot:                    1
        SP state:               Online
        Current Master KeyID:
a3:d7:57:c7:54:66:65:05:61:7a:35:2c:59:af:a5:dc
        Alternate Master KeyID:
e9:e4:3a:f8:bc:4e:75:44:81:35:b8:90:d0:1f:6f:4d
        HA Cluster Membership:  hacDcx2
        EE Attributes:
            Media Type     :    DISK
    EE Slot:                    3
        SP state:               Online
        Current Master KeyID:
a3:d7:57:c7:54:66:65:05:61:7a:35:2c:59:af:a5:dc
        Alternate Master KeyID:
e9:e4:3a:f8:bc:4e:75:44:81:35:b8:90:d0:1f:6f:4d
        No HA cluster membership
        EE Attributes:
            Media Type     :    DISK
EE Slot:                    10
        SP state:               Online
        Current Master KeyID:
a3:d7:57:c7:54:66:65:05:61:7a:35:2c:59:af:a5:dc
        Alternate Master KeyID:
e9:e4:3a:f8:bc:4e:75:44:81:35:b8:90:d0:1f:6f:4d
```

```
        No HA cluster membership
        EE Attributes:
            Media Type     :    DISK
    EE Slot:                    12
        SP state:               Online
        Current Master KeyID:
a3:d7:57:c7:54:66:65:05:61:7a:35:2c:59:af:a5:dc
        Alternate Master KeyID:
e9:e4:3a:f8:bc:4e:75:44:81:35:b8:90:d0:1f:6f:4d
        HA Cluster Membership:  hacDcx3
        EE Attributes:
            Media Type     :    DISK
```

Zoning considerations

When encryption is implemented, frames sent between a host and a target LUN are redirected to a virtual target within an encryption switch or blade. Redirection zones are created to route these frames. When redirection zones are in effect, direct access from host to target should not be allowed to prevent data corruption.

Set zone hosts and targets together before configuring them for encryption. Redirection zones are automatically created to redirect the host-target traffic through the encryption engine, but redirection zones can only be created if the host and target are already zoned.

Setting default zoning to no access

Initially, default zoning for all Brocade Encryption Switches is set to All Access. The All Access setting allows the Brocade Encryption Switch or DCX Backbone chassis to join the fabric and be discovered before zoning is applied. If there is a difference in this setting within the fabric, the fabric will segment.

Before committing an encryption configuration in a fabric, default zoning must be set to No Access within the fabric. The No Access setting ensures that no two devices on the fabric can communicate with one another without going through a regular zone or a redirection zone.

1. Check the default zoning setting. Commonly, it will be set to All Access.

   ```
   switch:admin> defzone --show
   Default Zone Access Mode
   committed - All Access
   transaction - No Transaction
   ```

2. From any configured primary FCS switch, change the default zoning setting to No Access.

   ```
   switch:admin> defzone --noaccess
   switch:admin> cfgfsave
   ```

   The change will be applied within the entire fabric.

Frame redirection zoning

Name Server-based frame redirection enables the Brocade Encryption Switch or blade to be deployed transparently to hosts and targets in the fabric.

NS-based frame redirection is enabled as follows:

- You first create a zone that includes host (H) and target (T). This may cause temporary traffic disruption to the host.
- You then create a CryptoTarget container for the target and configure the container to allow access to the initiator.
- When you commit the transaction, a special zone called a “redirection zone” is generated automatically. The redirection zone includes the host (H), the virtual target (VT), the virtual initiator (VI), and the target (T).
- When configuring multi-path LUNs, do not commit the CryptoTarget container configuration before you have performed the following steps in sequence to prevent data corruption. Refer to the section “Configuring a multi-path Crypto LUN” on page 198 for more information.
  - Complete all zoning for ALL hosts that should gain access to the targets.
  - Complete the CryptoTarget container configuration for ALL target ports in sequence, including adding the hosts that should gain access to these targets.

Host-target zoning must precede any CryptoTarget configuration.

NOTE
To enable frame redirection, the host and target edge switches must run Fabric OS v6.1.1 and Fabric OS v5.3.1.b or later firmware to ensure host and target connectivity with legacy platforms. In McDATA fabrics, the hosts and the switches hosting the targets require firmware versions M-EOSc 9.8 and M-EOSn 9.8 or later.

Creating an initiator - target zone

NOTES:

- NWWN based zoning of initiator and targets is not supported with Frame redirection.
- The Initiator-Target zone should be created before you create the container. Otherwise, the frame redirection zone creation for the Initiator-Target pair will fail during a commit.

1. Log in to the group leader as Admin or FabricAdmin.
2. Determine the initiator PWWN. Enter the **nsshow** command to view the devices connected to this switch. In the following example, the port name 10:00:00:00:c9:2b:c9:3a is the initiator PWWN.

```
FabricAdmin:switch> nsshow
{
 Type Pid    COS PortName                NodeName                TTL(sec)
 N 010600; 2,3;10:00:00:00:c9:2b:c9:3a;20:00:00:00:c9:2b:c9:3a; na
 NodeSymb: [35] "Emulex LP9002 FV3.82A1 DV5-4.81A4 "
 Fabric Port Name: 20:06:00:05:1e:41:9a:7e
 Permanent Port Name: 10:00:00:00:c9:2b:c9:3a
 Port Index: 6
 Share Area: No
 Device Shared in Other AD: No
 Redirect: No
 The Local Name Server has 1 entry }
```

The **nsshow** command shows all devices on the switch, and the output can be lengthy. To retrieve only the initiator PWWN, do a pattern search of the output based on the initiator Port ID (a hex number). In the following example, The PID is 010600, where 01 indicates the domain and 06 the port number.

```
FabricAdmin:switch> nsshow | grep 0106
 N 010600; 2,3;10:00:00:00:c9:2b:c9:3a;20:00:00:00:c9:2b:c9:3a; na
```

3. Determine the target PWWN. Enter the **nsscamshow** command to review the remote switch information. In the following example, the port name 20:0c:00:06:2b:0f:72:6d is the target PWWN.

```
FabricAdmin:switch> nscamshow
nscamshow for remote switches:
Switch entry for 2
  state rev  owner
  known v611 0xfffc01
  Device list: count 13
  Type Pid COS    PortName                 NodeName
  NL   0208d3; 3;20:0c:00:06:2b:0f:72:6d;20:00:00:06:2b:0f:72:6d;
    FC4s: FCP
    PortSymb: [55] "LSI7404XP-LC BR A.1 03-01081-02D FW:01.03.06 Port 1 "
    Fabric Port Name: 20:08:00:05:1e:34:e0:6b
    Permanent Port Name: 20:0c:00:06:2b:0f:72:6d
    Port Index: 8
    Share Area: No
    Device Shared in Other AD: No
    Redirect: No
```

Alternately use **nscamshow | grep** *target PID* to obtain the target PWWN only.

```
FabricAdmin:switch> nscamshow | grep 0208
NL   0208d3; 3;20:0c:00:06:2b:0f:72:6d;20:00:00:06:2b:0f:72:6d;
```

4. Create a zone that includes the initiator and a target. Enter the **zonecreate** command followed by a zone name, the initiator PWWN and the target PWWN.

```
FabricAdmin:switch> zonecreate itzone, "10:00:00:00:c9:2b:c9:3a; \
20:0c:00:06:2b:0f:72:6d"
```

5. Create a zone configuration that includes the zone you created in step 4. Enter the **cfgcreate** command followed by a configuration name and the zone member name.

```
FabricAdmin:switch> cfgcreate itcfg, itzone
```

6. Enable the zone configuration.

```
FabricAdmin:switch> cfgenable itcfg
You are about to enable a new zoning configuration.
This action will replace the old zoning configuration with the
current configuration selected.
Do you want to enable 'itcfg' configuration (yes, y, no, n): [no] y
zone config"itcfg" is in effect
Updating flash ...
```

7. Create a zone that includes the initiator and a LUN target. Enter the **zonecreate** command followed by a zone name, the initiator PWWN and the target PWWN.

```
FabricAdmin:switch> zonecreate itzone, "10:00:00:00:c9:2b:c9:3a; \
20:0c:00:06:2b:0f:72:6d"
```

8. Create a zone configuration that includes the zone you created in step 4. Enter the **cfgcreate** command followed by a configuration name and the zone member name.

```
FabricAdmin:switch> cfgcreate itcfg, itzone
```

9. Enable the zone configuration.

```
FabricAdmin:switch> cfgenable itcfg
You are about to enable a new zoning configuration.
This action will replace the old zoning configuration with the
current configuration selected.
Do you want to enable 'itcfg' configuration (yes, y, no, n): [no] y
zone config"itcfg" is in effect
Updating flash ...
```

CryptoTarget container configuration

A CryptoTarget container is a configuration of virtual devices created for each target port hosted on a Brocade Encryption Switch or FS8-18 blade. The container holds the configuration information for a single target, including associated hosts and LUN settings. A CryptoTarget container interfaces between the encryption engine, the external storage devices (targets), and the initiators (hosts) that can access the storage devices through the target ports. Virtual devices redirect the traffic between host and target/LUN to encryption engines so they can perform cryptographic operations.

Although an encryption engine can host more than one container for each target, it is not recommended.

Virtual targets: Any given physical target port is hosted on one encryption switch or blade. If the target LUN is accessible from multiple target ports, each target port is hosted on a separate encryption switch or blade. There is a one-to-one mapping between virtual target and physical target to the fabric whose LUNs are being enabled for cryptographic operations.

Virtual initiators: For each physical host configured to access a given physical target LUN, a virtual initiator (VI) is generated on the encryption switch or blade that hosts the target port. If a physical host has access to multiple targets hosted on different encryption switches or blades, you must configure one virtual initiator on each encryption switch or blade that is hosting one of the targets. The mapping between physical host and virtual initiator in a fabric is one-to-*n,* where *n* is the number of encryption switches or blades that are hosting targets.

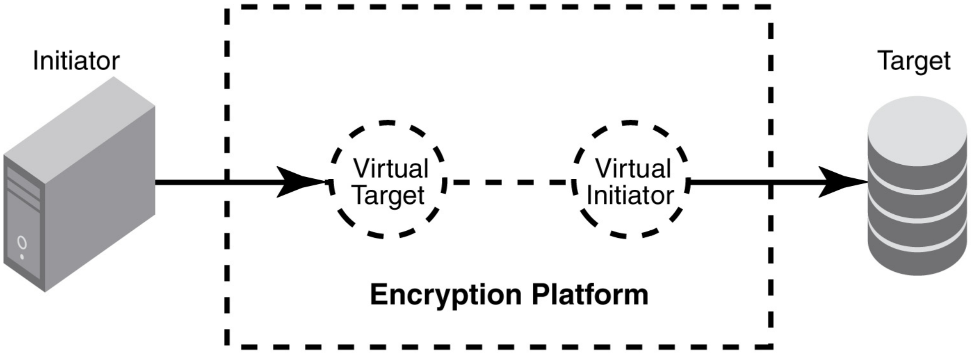


FIGURE 91 Relationship between initiator, virtual target, virtual initiator and target

CAUTION

When configuring a LUN with multiple paths, there is a considerable risk of ending up with potentially catastrophic scenarios where different policies exist for each path of the LUN, or a situation where one path ends up being exposed through the encryption switch and another path has direct access to the device from a host outside the secured realm of the encryption platform. Failure to follow correct configuration procedures for multi-path LUNs results in data corruption. If you are configuring multi-path LUNs as part of an HA cluster or DEK cluster or as a stand-alone LUN accessed by multiple hosts, follow the instructions described in the section “Configuring a multi-path Crypto LUN” on page 198.

LUN rebalancing when hosting both disk and tape targets

If you are currently using encryption and running Fabric OS v6.3.x or earlier, you are hosting tape and disk target containers on different encryption switches or blades. Beginning with Fabric OS v6.4, disk and tape target containers can be hosted on the same switch or blade. Hosting both disk and tape target containers on the same switch or blade may result in a drop in throughput, but it can reduce cost by reducing the number of switches or blades needed to support encrypted I/O in environments that use both disk and tape.

The throughput drop can be mitigated by rebalancing the tape and disk target containers across the encryption engine. This ensures that the tape and disk target containers are distributed within the encryption engine for maximum throughput.

All nodes within an encryption group must be upgraded to Fabric OS v6.4 or a later release to support hosting disk and tape target containers on the same encryption engine. If any node within an encryption group is running an earlier release, disk and tape containers must continue to be hosted on separate encryption engines.

During rebalancing operations, be aware of the following:

- You may notice a slight disruption in Disk I/O. In some cases, manual intervention may be needed.
- Backup jobs to tapes may need to be restarted after rebalancing completes.

To determine if rebalancing is recommended for an encryption engine, check the encryption engine properties. Beginning with Fabric OS v6.4, a field is added that indicates whether or not rebalancing is recommended.

You may be prompted to rebalance during the following operations:

- When adding a new disk or tape target container.
- When removing an existing disk or tape target container.
- After failover to a backup encryption engine in an HA cluster.
- After an failed encryption engine in an HA cluster is recovered, and failback processing has taken place.

To rebalance an encryption engine, do the following.

1. Log in to the switch as Admin or FabricAdmin.
2. Issue the **cryptocfg --show -localEE** command.
3. Look for **Rebalance recommended** under **EE Attributes** in the output.
4. If rebalancing is recommended, issue the **cryptocfg --rebalance** command.

Gathering information

Before you begin, have the following information ready:

- The switch WWNs of all nodes in the encryption group. Use the **cryptocfg --show -groupmember -all** command to gather this information.
- The port WWNs of the targets whose LUNs are being enabled for data-at-rest encryption.
- The port WWNs of the hosts (initiators) which should gain access to the LUNs hosted on the targets.

Any given target may have multiple ports through which a given LUN is accessible and the ports are connected to different fabrics for redundancy purposes. Any given target port through which the LUNs are accessible must be hosted on only one Encryption switch (or pair in case of HA deployment). Another such target port should be hosted on a different encryption switch either in the same fabric or in a different fabric based on host MPIO configuration.

A given host port through which the LUNs are accessible is hosted on the same encryption switch on which the target port (CryptoTarget container) of the LUNs is hosted.

NOTE
It is recommended you complete the encryption group and HA cluster configuration before configuring the CryptoTarget containers.

Creating a CryptoTarget container

1. Log in to the group leader as Admin or FabricAdmin.
2. Enter the **cryptocfg --create -container** command. Specify the type of container, (disk or tape), followed by a name for the CryptoTarget container, the encryption engine's node WWN, and the target's Port WWN and node WWN. Provide a slot number if the encryption engine is a blade.
    - The CryptoTarget container name can be up to 31 characters in length and may include any alphanumeric characters, hyphens, and underscore characters.
    - You may add initiators at this point or after you create the container.

The following example creates a disk container named my_disk_tgt1. The initiator is added in step 3.

```
FabricAdmin:switch> cryptocfg --create -container disk my_disk_tgt \
10:00:00:00:05:1e:41:9a:7e 20:0c:00:06:2b:0f:72:6d 20:00:00:06:2b:0f:72:6d
Operation Succeeded
```

3. Add an initiator to the CryptoTarget container. Enter the **cryptocfg** **--add** **-initiator** command followed by the initiator port WWN and the node WWN.

   Note that the initiator port WWN must also be added to the LUN when the LUN is added to the CryptoTarget container.

```
FabricAdmin:switch> cryptocfg --add -initiator my_disk_tgt \
10:00:00:00:c9:2b:c9:3a 20:00:00:00:c9:2b:c9:3a
Operation Succeeded
```

4. Commit the transaction. The commit operation creates the virtual devices and the redirection zone that routes traffic through these devices.

```
FabricAdmin:switch> cryptocfg --commit
Operation Succeeded
```

CAUTION

When configuring a multi-path LUN, you must complete the CryptoTarget container configuration for ALL target ports in sequence and add the hosts that should gain access to these ports before committing the container configuration. Failure to do so results in data corruption. Refer to the section "Configuring a multi-path Crypto LUN" on page 198 for specific instructions.

5. Display the CryptoTarget container configuration. The virtual initiator and virtual target have been created automatically upon commit, and there are no LUNs configured yet.

```
FabricAdmin:switch> cryptocfg --show -container my_disk_tgt -cfg
Container name:       my_disk_tgt
Type:                 disk
EE node:              10:00:00:05:1e:41:9a:7e
EE slot:              0
Target:               20:0c:00:06:2b:0f:72:6d 20:00:00:06:2b:0f:72:6d
VT:                   20:00:00:05:1e:41:4e:1d 20:01:00:05:1e:41:4e:1d
Number of host(s):    1
Configuration status: committed
Host:                 10:00:00:00:c9:2b:c9:3a 20:00:00:00:c9:2b:c9:3a
VI:                   20:02:00:05:1e:41:4e:1d 20:03:00:05:1e:41:4e:1d
Number of LUN(s):     0
Operation Succeeded
```

6. Display the redirection zone. It includes the host, the target, the virtual initiator, and the virtual target.

```
FabricAdmin:switch> cfgshow
Defined configuration:
cfg:   itcfg          itzone
cfg:   r_e_d_i_r_c__fg
            red_1109_brcd200c00062b0f726d200200051e414e1d; red_______base
zone:  itzone 10:00:00:00:c9:2b:c9:3a; 20:0c:00:06:2b:0f:72:6d
```

```
zone:  red_1109_brcd200c00062b0f726d200200051e414e1d
            10:00:00:00:c9:2b:c9:3a; 20:0c:00:06:2b:0f:72:6d;
            20:02:00:05:1e:41:4e:1d; 20:00:00:05:1e:41:4e:1d
zone: red_______base
       00:00:00:00:00:00:00:01; 00:00:00:00:00:00:00:02;
       00:00:00:00:00:00:00:03; 00:00:00:00:00:00:00:04
Effective configuration:
cfg:   itcfg
zone:  itzone  10:00:00:00:c9:2b:c9:3a
               20:0c:00:06:2b:0f:72:6d
```

NOTE
You may view the frame redirection zone with the **cfgshow** command, but you cannot use the zone for any other applications that use frame redirection. Do not perform any further operations with this zone, such as deleting the zone or adding the zone to a different configuration. Such operations may result in disruptive behavior, including data corruption on the LUN.

Removing an initiator from a CryptoTarget container

You may remove one or more initiators from a given CryptoTarget container. This operation removes the initiators' access to the target port.

If the initiator has access to multiple targets and you wish to remove access to all targets, follow the procedure described to remove the initiator from every CryptoTarget container that is configured with this initiator.

NOTE
Stop all traffic between the initiator you intend to remove and its respective target ports. Failure to do so results in I/O failure between the initiator and the target port.

1. Log in to the group leader as Admin or FabricAdmin.
2. Enter the **cryptocfg --remove -initiator** command. Specify the CryptoTarget container name followed by one or more initiator port WWNs. The following example removes one initiator from the CryptoTarget container "my_disk_tgt".

```
FabricAdmin:switch> cryptocfg --rem -initiator my_disk_tgt
10:00:00:00:c9:2b:c9:3a
Operation Succeeded
```

3. Commit the transaction.

```
FabricAdmin:switch> cryptocfg --commit
Operation Succeeded
```

CAUTION

When configuring a multi-path LUN, you must remove all initiators from all CryptoTarget containers in sequence before committing the transaction. Failure to do so may result in a potentially catastrophic situation where one path ends up being exposed through the encryption switch and another path has direct access to the device from a host outside the protected realm of the encryption platform. Refer to the section "Configuring a multi-path Crypto LUN" on page 198 for more information.

Deleting a CryptoTarget container

You may delete a CryptoTarget container to remove the target port from a given encryption switch or blade. Deleting a CryptoTarget container removes the virtual target and all associated LUNs from the fabric.

Before deleting a container, be aware of the following:

- Stop all traffic to the target port for which the CryptoTarget container is being deleted. Failure to do so will cause data corruption (a mix of encrypted data and cleartext data will be written to the LUN).
- Deleting a CryptoTarget container while a rekey or first-time encryption session causes all data to be lost on the LUNs that are being rekeyed. Ensure that no rekey or first-time encryption sessions are in progress before deleting a container. Use the **cryptocfg --show -rekey -all** command to determine the runtime status of the session. If for some reason, you need to delete a container while rekeying, when you create a new container, be sure the LUNs added to the container are set to **cleartext**. You can then start a new rekey session on clear text LUNs.

1. Log in to the group leader as Admin or FabricAdmin.
2. Enter the **cryptocfg --delete -container** command followed by the CryptoTarget container name. The following example removes the CryptoTarget container "my_disk_tgt".

```
FabricAdmin:switch> cryptocfg --delete -container my_disk_tgt
Operation Succeeded
```

3. Commit the transaction.

```
FabricAdmin:switch> cryptocfg --commit
Operation Succeeded
```

CAUTION

When configuring a multi-path LUN, you must remove all necessary CryptoTarget containers in sequence before committing the transaction. Failure to do so may result in a potentially catastrophic situation where one path ends up being exposed through the encryption switch and another path has direct access to the device from a host outside the protected realm of the encryption platform. Refer to the section "Configuring a multi-path Crypto LUN" on page 198 for more information.

Moving a CryptoTarget container

You can move a CryptoTarget container from one encryption engine to another. The encryption engines must be part of the same fabric and the same encryption group, and the encryption engines must be online for this operation to succeed. This operation permanently transfers the encryption engine association of a given CryptoTarget container from an existing encryption engine to an alternate encryption engine.

NOTE
If a CryptoTarget container is moved in a configuration involving FCR, the LSAN zones and manually created redirect zones will need to be reconfigured with new VI and VT WWNs. Refer to the section "Deployment in Fibre Channel routed fabrics" on page 217 for instructions on configuring encryption in an FCR deployment scenario.

1. Log in to the group leader as Admin or FabricAdmin.
2. Enter the **cryptocfg --move -container** command followed by the CryptoTarget container name and the node WWN of the encryption engine to which you are moving the CryptoTarget container. Provide a slot number if the encryption engine is a blade.

```
FabricAdmin:switch> cryptocfg --move -container my_disk_tgt \
10:00:00:05:1e:53:4c:91
Operation Succeeded
```

3. Commit the transaction.

```
FabricAdmin:switch> cryptocfg --commit
Operation Succeeded
```

Crypto LUN configuration

A Crypto LUN is the LUN of a target disk or tape storage device that is enabled for and capable of data-at-rest encryption. Crypto LUN configuration is done on a per-LUN basis. You configure the LUN for encryption by explicitly adding the LUN to the CryptoTarget container and turning on the encryption property and policies on the LUN. Any LUN of a given target that is not enabled for encryption must still be added to the CryptoTarget container with the **cleartext** policy option.

- The general procedures described in this section apply to both disk and tape LUNs. The specific configuration procedures differ with regard to encryption policy and parameter setting.
- You configure the Crypto LUN on the group leader. You need the Admin or FabricAdmin role to perform LUN configuration tasks.
- With the introduction of Fabric OS 7.1.0, the maximum number of uncommitted configuration changes per disk LUN (or maximum paths to a LUN) is 512 transactions. This change in commit limit is applicable only when using BNA.The commit limit when using the CLI remains unchanged at 25.
- There is a maximum of eight tape LUNs per Initiator in a container. The maximum number of uncommitted configuration changes per tape LUN remains unchanged at eight.

CAUTION

When configuring a LUN with multiple paths (which means the LUN is exposed and configured on multiple CryptoTarget containers located on the same Encryption switch or blade, or on different encryption switches or blades), the same LUN policies must be configured on all LUN paths. Failure to configure all LUN paths with the same LUN policies results in data corruption. If you are configuring multi-path LUNs as part of a HA cluster or DEK cluster or as a stand-alone LUN accessed by multiple hosts, follow the instructions described in the section "Configuring a multi-path Crypto LUN" on page 198.

Discovering a LUN

When adding a LUN to a CryptoTarget container, you must specify a LUN Number. The LUN Number needed for configuring a given Crypto LUN is the LUN Number as exposed to a particular initiator.

The Brocade encryption platform provides LUN discovery services through which you can identify the exposed LUN number for a specified initiator. If you already know the exposed LUN numbers for the various initiators accessing the LUN, you may skip the LUN discovery step and directly configure the Crypto LUN.

1. Log in to the group leader as Admin or FabricAdmin.
2. Enter the **cryptocfg --discoverLUN** command followed by the CryptoTarget container Name.

```
FabricAdmin:switch> cryptocfg --discoverLUN my_disk_tgt
Container name: my_disk_tgt
Number of LUN(s): 1
Host: 10:00:00:00:c9:2b:c9:3a
LUN number: 0x0
LUN serial number: 200000062B0F726D0C000000
Key ID state: Key ID not available
Key ID: 3a:21:6a:bd:f2:37:d7:ea:6b:73:f6:19:72:89:c6:4f
```

CAUTION

When configuring a LUN with multiple paths, perform the LUN discovery on each of the CryptoTarget containers for each of the paths accessing the LUN and verify that the serial number for these LUNs discovered from these CryptoTarget containers are the same. This indicates and validates that these CryptoTarget containers are indeed paths to the same LUN. Refer to the section "Configuring a multi-path Crypto LUN" on page 198 for more information.

Configuring a Crypto LUN

You configure a Crypto LUN by adding the LUN to the CryptoTarget container and enabling the encryption property on the Crypto LUN. The LUNs of the target that are not enabled for encryption must still be added to the CryptoTarget container with the **cleartext** policy option.

You can add a single LUN to a CryptoTarget container, or you can add multiple LUNs by providing a range of LUN Numbers. When adding a single LUN, you can either provide a 16-bit (2 byte) hex value of the LUN Number, for example, 0x07. Alternately you can provide a 64-bit (8 byte) value in WWN or LUN ID format, for example, 00:07:00:00:00:00:00:00. When adding a range of LUN Numbers, you may use two byte hex values or decimal numbers.

LUN configurations and modifications must be committed to take effect. The commit limit when using the CLI is 25. If the number of paths for a LUN exceeds the limit, then more than one transaction must be sent. Attempts to commit configurations or modifications that exceed the maximum commit allowed will fail with a warning. There is also a five-second delay before the commit operation takes effect. In addition to the commit limits, make sure the LUNs in previously committed LUN configurations and LUN modifications have a LUN state of **Encryption Enabled** before creating and committing another batch of LUN configurations or LUN modifications.

NOTE
There is a maximum of 512 disk LUNs per Initiator in a container. With the introduction of Fabric OS 7.1.0, the maximum number of uncommitted configuration changes per disk LUN (or maximum paths to a LUN) is 512 transactions. This change in commit limit is applicable only when using BNA.The commit limit when using the CLI remains unchanged at 25.

NOTE
The maximum of number of tape LUNs that can be added or modfied in a single commit operation remains unchanged at eight.

The device type (disk or tape) is set at the CryptoTarget container level. You cannot add a tape LUN to a CryptoTarget container of type “disk” and vice versa.

It is recommended that you configure the LUN state and encryption policies at this time. You can add these settings later with the **cryptocfg --modify -LUN** command, but not all options are modifiable. Refer to the section “Crypto LUN parameters and policies” on page 169 for LUN configuration parameters. Refer to the section “Creating a tape pool” on page 196 for tape pool policy parameters.

NOTE
If you are using VMware virtualization software or any other configuration that involves mounted file systems on the LUN, you must enable first-time encryption at the time when you create the LUN by setting the **enable_encexistingdata** option with the **--add -LUN** command. Failure to do so permanently disconnects the LUN from the host and causes data to be lost and unrecoverable.

1. Log in to the group leader as Admin or FabricAdmin.
2. Enter the **cryptocfg --add -LUN** command followed by the CryptoTarget container Name, the LUN number or a range of LUN numbers, the PWWN and NWWN of the initiators that should be able to access the LUN. The following example adds a disk LUN enabled for encryption.

```
FabricAdmin:switch> cryptocfg --add -LUN my_disk_tgt 0x0 \
10:00:00:00:c9:2b:c9:3a 20:00:00:00:c9:2b:c9:3a -encrypt
Operation Succeeded
```

3. Commit the configuration.

```
FabricAdmin:switch> cryptocfg --commit
Operation Succeeded
```

CAUTION

When configuring a LUN with multiple paths, do not commit the configuration before you have added all the LUNs with identical policy settings and in sequence to each of the CryptoTarget containers for each of the paths accessing the LUNs. Failure to do so results in data corruption. Refer to the section "Configuring a multi-path Crypto LUN" on page 198.

4. Display the LUN configuration. The following example shows default values.

```
FabricAdmin:switch> cryptocfg --show -LUN my_disk_tgt0 \
10:00:00:00:c9:2b:c9:3a -cfg
EE node: 10:00:00:05:1e:41:9a:7e
EE slot: 0
Target: 20:0c:00:06:2b:0f:72:6d 20:00:00:06:2b:0f:72:6d
VT: 20:00:00:05:1e:41:4e:1d 20:01:00:05:1e:41:4e:1d
Number of host(s): 1
Configuration status: committed
Host: 10:00:00:00:c9:2b:c9:3a 20:00:00:00:c9:2b:c9:3a
VI: 20:02:00:05:1e:41:4e:1d 20:03:00:05:1e:41:4e:1d
LUN number: 0x0
LUN type: disk
LUN status: 0
Encryption mode: encrypt
Encryption format: native
Encrypt existing data: enabled
Rekey: disabled
Key ID: not available
Operation Succeeded
```

Crypto LUN parameters and policies

Table 6 shows the encryption parameters and policies that can be specified for a disk or tape LUN, during LUN configuration (with the **cryptocfg --add -LUN** command). Some policies are applicable only to disk LUNs, and some policies are applicable only to tape LUNs. It is recommended that you plan to configure all the LUN state and encryption policies with the **cryptocfg --add -LUN** command. You can use the **cryptocfg --modify -LUN** command to change some of the settings, but not all options can be modified.

NOTE
LUN policies are configured at the LUN level, but apply to the entire HA or DEK cluster. For multi-path LUNs that are exposed through multiple target ports and thus configured on multiple CryptoTarget containers on different encryption engines in an HA cluster or DEK cluster, the same LUN policies must be configured. Failure to do so results in unexpected behavior and may lead to data corruption.

The tape policies specified at the LUN configuration level take effect if you do not create tape pools or configure policies at the tape pool level. The Brocade encryption solutions supports up to a 1 MB block size for tape encryption. Also, the Logical Block Address (LBA) 0 block size (I/O size from the host) must be at least 1 K less than the maximum supported backend block size (usually 1 MB). This is typically the case, as label operations are small I/O operations. If this support requirement is not met, the Brocade encryption solution will not allow the backup operation to start to that tape.

NOTE
LBA 0 is not encrypted. Data sent to this block address is always sent as clear text.

TABLE 6 LUN parameters and policies

Policy name	Command parameters	Description
LUN state Disk LUN: yes Tape LUN: No Modify? No	–**lunstate** encrypted | cleartext	Sets the Encryption state for the LUN. Valid values are: • **cleartext** - Default LUN state. Refer to policy configuration considerations for compatibility with other policy settings. • **encrypted** - Metadata on the LUN containing the key ID of the DEK that was used for encrypting the LUN is used to retrieve the DEK from the key vault. DEKs are used for encrypting and decrypting the LUN.
Key ID Disk LUN: yes Tape LUN: No Modify? No	–**keyID** *Key_ID*	Specifies the key ID. Use this option *only* if the LUN was encrypted but does not include the metadata containing the key ID for the LUN. This is a rare case for LUNs encrypted in **Native** (Brocade) mode.
Encryption format Disk LUN: yes Tape LUN: yes Modify? Yes	–**encryption_format** native	Sets the encryption format. Valid values are: • **Native** - The LUN is encrypted or decrypted using the Brocade encryption format (metadata format and algorithm). This is the default setting.
Encryption policy Disk LUN: yes Tape LUN: Yes Modify? Yes	–**encrypt** | –**cleartext**	Enables or disables a LUN for encryption. Valid values are: • **cleartext** - Encryption is disabled. This is the default setting. When the LUN policy is set to **cleartext** the following policy parameters are invalid and generate errors when executed: –**enable_encexistingdata** –**enable_rekey**, and –**key_lifespan**. • **encrypt** - The LUN is enabled to perform encryption.
Existing data encryption Disk LUN: yes Tape LUN: No Modify? Yes	**-enable_encexistingdata** | **-disable_encexistingdata**	Specifies whether or not existing data on the LUN should be encrypted. By default, encryption of existing data is disabled. Encryption policy must be set to –**enable_encexistingdata**, and the LUN state must be set to **cleartext** (default). If the encryption policy is **cleartext**, the existing data on the LUN will be overwritten.
Rekey policy Disk LUN: yes Tape LUN: No Modify? Yes	**-enable_rekey** *time_period <days>* | –**disable_rekey**	Enables or disables the auto rekeying feature on a specified disk LUN. This policy is not valid for tape LUNs. By Default, the automatic rekey feature is disabled. Enabling automatic rekeying is valid only if the LUN policy is set to –**encrypt**. You must specify a time period in days when enabling Auto Rekey to indicate the interval at which automatic rekeying should take place.
Key lifespan Disk LUN: No Tape LUN: Yes Modify? Disks only. Tape: No	–**key_lifespan** *time_in_days* | **none**	Specifies the life span of the encryption key in days. The key will expire after the specified number of days. Accepted values are integers from 1 to 2982616. The default value is none, which means the key does not expire. On tape LUNs, the key life span cannot be modified after it is set.

TABLE 6 LUN parameters and policies (Continued)

Policy name	Command parameters	Description
Write Early Ack Disk LUN: No Tape LUN: Yes Modify? Tape Only. Disk: No	-write_early_ack disable|enable	Specifies the Tape Write pipelining mode of the LUN. Two Write Pipelining modes are supported: • **disable** - Early acknowledgement of commands (internal buffering) for a tape lun is disabled. • **enable** - Early acknowledgement of commands (internal buffering) for a tape lun is enabled. The default value is enable.
Read Ahead Disk LUN: No Tape LUN: Yes Modify? Tape Only. Disk: No	–read_ahead disable | enable	Specifies the Tape Read Ahead mode of the LUN. Two Read Ahead modes are supported: • **disable** - The LUN disables the Tape read ahead and Tape LUN will be operated in unbuffered mode. • **enable** - The LUN enables the Tape read ahead and Tape LUN will be operated in buffered mode. The default value is enable.
NewLUN Disk LUN: Yes Tape LUN: No Modify? No		Specifies the LUN to be a configured as a replication LUN for SRDF, TimeFinder or RecoverPoint.

Configuring a tape LUN

This example shows how to configure a tape storage device. The basic setup procedure is the same as for disk devices. Only a subset of configuration options and policy settings are available for tape LUNs. Refer to Table 6 on page 170 for tape LUN configuration options.

1. Create a zone that includes the initiator (host) and the target port. Refer to the section "Creating an initiator - target zone" on page 158 for instructions.

2. Create a CryptoTarget container of type **tape**. Refer to the section "Creating a CryptoTarget container" on page 162 for instructions.

   a. Create the container, allowing the encryption format to default to Native.

   ```
   FabricAdmin:switch> cryptocfg --create -container tape my_tape_tgt \
   10:00:00:05:1e:41:9a:7e 20:0c:00:06:2b:0f:72:6d 20:00:00:06:2b:0f:72:6d
   Operation Succeeded
   ```

   b. Add an initiator to the CryptoTarget container "my_tape_tgt".

   ```
   FabricAdmin:switch> cryptocfg --add -initiator my_tape_tgt \
   10:00:00:00:c9:2b:c9:3a 20:00:00:00:c9:2b:c9:3a
   Operation Succeeded
   ```

   c. Commit the transaction.

   ```
   FabricAdmin:switch> cryptocfg --commit
   Operation Succeeded
   ```

3. Configure the Crypto tape LUN. Refer to the section "Configuring a Crypto LUN" on page 168 for instructions.

a. Discover the LUN.

```
FabricAdmin:switch> cryptocfg --discoverLUN my_tape_tgt
Container name:         my_tape_tgt
Number of LUN(s):       1
Host:                  10:00:00:00:c9:2b:c9:3a
LUN number:            0x0
LUN serial number:
Key ID state:          Key ID not Applicable
```

b. Add the LUN to the tape CryptoTarget container. The following example enables the LUN for encryption. There is a maximum of eight tape LUNs per Initiator in a container.

```
FabricAdmin:switch> cryptocfg --add -LUN my_tape_tgt 0x0 \
10:00:00:00:c9:2b:c9:3a 20:00:00:00:c9:2b:c9:3a -encrypt
Operation Succeeded
```

NOTE
When changing the tape LUN policy from **encrypt** to **cleartext** or from **cleartext** to **encrypt,** or the encryption format from Brocade **native** to **DF-compatible** while data is being written to or read from a tape backup device, the policy change is not enforced until the current process completes and the tape is unmounted, rewound, or overwritten. Refer to the section "Impact of tape LUN configuration changes" on page 175 for more information.

c. Commit the configuration.

```
FabricAdmin:switch> cryptocfg --commit
Operation Succeeded
```

d. Display the LUN configuration.

```
FabricAdmin:switch> cryptocfg --show -LUN my_tape_tgt 0x0 \
10:00:00:00:c9:2b:c9:3a -cfg
EE node:               10:00:00:05:1e:41:9a:7e
EE slot:               0
Target:                20:0c:00:06:2b:0f:72:6d 20:00:00:06:2b:0f:72:6d
VT:                    20:00:00:05:1e:41:4e:1d 20:01:00:05:1e:41:4e:1d
Number of host(s):     1
Configuration status: committed
Host:                  21:00:00:e0:8b:89:9c:d5 20:00:00:e0:8b:89:9c:d5
VI:                    10:00:00:00:c9:2b:c9:3a 20:03:00:05:1e:41:4e:31
LUN number:            0x0
LUN type:              tape
LUN status:            0
Encryption mode:       encrypt
Encryption format:     DF_compatible
Tape type:             tape
Key life:              90 (day)
Volume/Pool label:
Operation succeeded.
```

Removing a LUN from a CryptoTarget container

You can remove a LUN from a given CryptoTarget container if it is no longer needed. Stop all traffic I/O from the initiators accessing the LUN before removing the LUN to avoid I/O failure between the initiators and the LUN. If the LUN is exposed to more than one initiator under different LUN Numbers, remove all exposed LUN Numbers.

1. Log in to the group leader as Admin or FabricAdmin.
2. Enter the **cryptocfg --remove -LUN** command followed by the CryptoTarget container name, the LUN Number, and the initiator PWWN.

   ```
   FabricAdmin:switch> cryptocfg --remove -LUN my_disk_tgt 0x0
   10:00:00:00:c9:2b:c9:3a
   Operation Succeeded
   ```

3. Commit the configuration with the **-force** option to completely remove the LUN and all associated configuration data in the configuration database. The data remains on the removed LUN in an encrypted state.

   ```
   FabricAdmin:switch> cryptocfg --commit -force
   Operation Succeeded
   ```

CAUTION

In case of multiple paths for a LUN, each path is exposed as a CryptoTarget container in the same encryption switch or blade or on different encryption switches or blades within the encryption group. In this scenario you must remove the LUNs from all exposed CryptoTarget containers before you commit the transaction. Failure to do so may result in a potentially catastrophic situation where one path ends up being exposed through the encryption switch and another path has direct access to the device from a host outside the protected realm of the encryption platform. Refer to the section "Configuring a multi-path Crypto LUN" on page 198 for more information.

Modifying Crypto LUN parameters

You can modify one or more policies of an existing Crypto LUN with the **cryptocfg --modify -LUN** command.

A maximum of 25 disk LUNs can be added or modified in a single commit operation through the CLI. Attempts to commit configurations or modifications that exceed the maximum commit allowed will fail with a warning. There is a five second delay before the commit operation takes effect.

Make sure the LUNs in previously committed LUN configurations and LUN modifications have a LUN state of **Encryption Enabled** before creating and committing another batch of LUN configurations or modifications.

The following example disables automatic rekeying operations on the disk LUN "my_disk_tgt."

1. Log in to the group leader as Admin or FabricAdmin.
2. Enter the **cryptocfg --modify -LUN** command followed by the CryptoTarget container name, the LUN Number, the initiator PWWN, and the parameter you want to modify.

   ```
   FabricAdmin:switch> cryptocfg --modify -LUN my_disk_tgt 0x0
   10:00:00:00:c9:2b:c9:3a -disable_rekey
   ```

```
Operation Succeeded
```

3. Commit the configuration.

```
FabricAdmin:switch> cryptocfg --commit
Operation Succeeded
```

CAUTION

When configuring a LUN with multiple paths, do not commit the configuration before you have modified all the LUNs with identical policy settings and in sequence for each of the CryptoTarget containers for each of the paths accessing the LUNs. Failure to do so results in data corruption. Refer to the section "Configuring a multi-path Crypto LUN" on page 198.

LUN modification considerations

Make sure you understand the ramifications of modifying LUN policy parameters (such as encrypt/cleartext) for LUNs that are online and already being utilized. The following restrictions apply when modifying LUN policy parameters for disk LUNs:

- When you change LUN policy from **encrypt** to **cleartext**, you wipe out all encrypted data stored on the LUN the next time data is written to that LUN. The following policy parameters are disabled: -**enable_encexistingdata**, -**enable_rekey**.
- When you change the LUN policy back to **encrypt**, for example, by force-enabling the LUN, **-enable_encexistingdata** and -**enable_rekey** are disabled by default, and you must configure both options again.
- When you add a LUN as **cleartext** and later you want to change the LUN policy from **cleartext** to **encrypt**, you must set the -**enable_encexistingdata** option. If you do not, all data on that LUN is lost, and cannot be recovered.

For tape LUNs, the -**enable_encexistingdata,** -**enable_rekey**, and -**key_lifespan** options are not valid and therefore cannot be modified. When you attempt to execute these parameters while modifying a tape LUN, the system returns an error. Disabling -**write_early ack** or -**read_ahead** for tape LUN will result in lower total throughput depending on the number of flows per encryption engine.

NOTE

Make sure all the outstanding backup and recovery operations on the media are completed before changing the LUN configuration.

For Disk LUNs -**write_early_ack** and -**read_ahead** are not valid and therefore cannot be modified. When you attempt to execute these parameters while modifying a disk LUN, the system returns an error.

Impact of tape LUN configuration changes

LUN-level policies apply when no policies are configured at the tape pool level. The following restrictions apply when modifying tape LUN configuration parameters:

- If you change a tape LUN policy from **encrypt** to **cleartext** or from **cleartext** to **encrypt** while data is written to or read from a tape backup device, the policy change is not enforced until the current process completes and the tape is unmounted, rewound, or overwritten. This mechanism prevents the mixing of cleartext data to cipher-text data on the tape.
- Make sure you understand the ramifications of changing the tape LUN encryption policy from **encrypt** to **cleartext** or from **cleartext** to **encrypt**.
- You cannot modify the key lifespan value. If you wish to modify the key lifespan, delete and recreate the LUN with a different key lifespan value. Key lifespan values only apply to native-mode pools.

Decommissioning LUNs

A disk device needs to be decommissioned when any of the following occur:

- The storage lease expires for an array, and devices must be returned or exchanged.
- Storage is reprovisioned for movement between departments.
- An array or device is removed from service.

In all cases, all data on the disk media must be rendered inaccessible. Device decommissioning deletes all information that could be used to recover the data, for example, information related to master key IDs and cache files.

After device decommissioning is performed, the following actions occur:

- Metadata on the LUN is erased and the reference is removed from cache on the Brocade Encryption Switch.
- The LUN state is shown as decommissioned in the key vault.
- The LUN is removed from the container.

NOTE
The key IDs that were used for encrypting the data are returned.

When a device decommission operation fails on the encryption group leader for any reason, the crypto configuration remains uncommitted until a user-initiated commit or a subsequent device decommission operation issued on the encryption group leader completes successfully. Device decommission operations should always be issued from a committed configuration. If not, the operation will fail with the error message **An outstanding transaction is pending in Switch/EG**. IF this happens, you can resolve the problems by committing the configuration from the encryption group leader.

Provided that the crypto configuration is not left uncommitted because of any crypto configuration changes or a failed device decommission operation issued on a encryption group leader node, this error message will not be seen for any device decommission operation issued serially on an encryption group member node. If more than one device decommission operation is tried in an encryption group from member nodes simultaneously, then this error message is transient and will go away after device decommission operation is complete. If the device decommissioning operation fails, retry the operation after some time has passed.

If a LUN is removed when undergoing decommission or is in a decommission failed state, or if a container hosting the LUN is deleted, you must use the -**force** option on the commit operation (**cryptocfg** --**commit** -**force**). Failure to do so causes the commit operation to fail and a decommission in progress error displays.

Upon a successful completion of a decommissioning operation, the LUN is deleted from all containers hosting it, and all active paths to the LUNs are lost.

NOTE
In a mixed encryption group consisting of nodes running Fabric OS 7.0.0 and an earlier Fabric OS version (for example, Fabric OS 6.4.2), the decommission operation will complete successfully and the LUNs will be removed from the hosted containers; however, the list of decommissioned key IDs might not be displayed correctly from all nodes in the encryption group. To resolve this, ensure that the Fabric OS version running on all nodes in an encryption group is the same version. Otherwise some of the **crypto** commands might not work as expected.

Complete the following procedure to decommission a disk LUN.

1. Log in as Admin or FabricAdmin to the node that hosts the container.
2. Enter the **cryptocfg** --**decommission** command.

   ```
   FabricAdmin:switch> cryptocfg --decommission -container disk_ct0 -initiator
   21:01:00:1b:32:29:5d:1c -LUN 0
   ```

3. Enter **cryptocfg** --**show** -**decommissionedkeyids** to obtain a list of all currently decommissioned key IDs to be deleted after decommissioning key IDs manually from the key vault.

   ```
   FabricAdmin:switch> cryptocfg --show -decommissionedkeyids
   ```

4. Enter the **cryptocfg** --**show** -**vendorspecific_keyid <key_id>** command to list the vendor-specific key information for a given key ID.

   ```
   FabricAdmin:switch> cryptocfg --show -vendorspecific_keyid
   AA:8B:91:B0:35:6F:DA:92:8A:72:B3:97:92:1B:CA:B4
   uuid = b7e07a6a-db64-40c2-883a-0bc6c4e923e6
   ```

5. Manually delete the listed key IDs from the key vault.
6. Enter the **cryptocfg** --**delete** -**decommissionedkeyids** command to purge all key IDs associated with a decommissioned LUN.

   ```
   FabricAdmin:switch> cryptocfg --delete -decommissionedkeyids
   ```

7. Enter the **cryptocfg** --**show** -**decommissionedkeyids** command to verify that the deleted key IDs are no longer listed.

   The cache is also cleared when **cryptocfg** --**zeroizeEE** is executed on the encryption engine.

NOTES:

- When a decommissioned LUN is reused and the decommissioned key IDs are listed using the **cryptocfg** --**show** -**decommissionedkeyids** command, the entire list of decommissioned key IDs since the first time the LUN was used is displayed.

- If you are running Fabric OS 7.1.0, and you want to downgrade to an earlier Fabric OS version, (for example, Fabric OS 7.0.x), after decommissioning a disk LUN, it is recommended that you remove the decommissioned key ID from the key vault *before* performing the downgrade. Otherwise, if the LUN is added back for encryption, the LUN will go to the disabled state as the key state is decommissioned in the key vault.

Decommissioning replicated LUNs

When trying to re-use primary R1 or secondary R2 replicated LUNs, you must first decommission the LUNs. When trying to re-use a decommissioned LUN, you must:

1. Delete the keys from the key vault.
2. Add the LUN back into the container as cleartext.
3. Modify the LUN to encrypted.

The following scenarios are provided:

- "Decommissioning primary R1 LUNs only"
- "Decommissioning mirror R2 LUNs only"
- "Decommissioning primary R1 and mirror R2 LUN pairs"

Decommissioning primary R1 LUNs only

To decommission the primary LUN and make the secondary LUN the primary LUN, complete the following steps. Failure to do so could result in the LUN state showing as Disabled.

1. Log in as Admin or FabricAdmin.
2. Split the R1/R2 sync.
3. Make the R2 LUN write-enabled.
4. Execute the **rekey** command on the R2 LUN.

   ```
   FabricAdmin:switch> cryptocfg --manual_rekey <crypto target container name>
   <LUN Num> <Initiator PWWN>
   ```

5. Decommission the primary LUN.

   ```
   FabricAdmin:switch> cryptocfg --decommission -container <container name>
   -initiator <initiator  PWWN> -LUN <lun number>
   ```

6. Display the decommissioned key IDs.

   ```
   FabricAdmin:switch> cryptocfg --show -decommissionedkeyids
   ```

7. Delete the respective key from the key vault. On the Brocade Encryption Switch, enter the following command.

   ```
   FabricAdmin:switch> cryptocfg --delete -decommissionedkeyids
   ```

NOTE
Failure to rekey the secondary LUN might result in loss of data on the secondary LUN after the primary LUN is decommissioned.

Decommissioning mirror R2 LUNs only

To decommission the secondary LUN, complete the following steps:

1. Log in as Admin or FabricAdmin.
2. Split the R1/R2 sync.
3. Make the R2 LUN write-enabled.
4. Decommission the R2 LUN.

```
FabricAdmin:switch> cryptocfg --decommission -container <container name>
-initiator <initiator  PWWN> -LUN <lun number>
```

NOTE
Do not delete the key from the key vault.

Decommissioning primary R1 and mirror R2 LUN pairs

To decommission both the primary and secondary LUNs, complete the following steps:

1. Log in as Admin or FabricAdmin.
2. Split the R1/R2 sync.
3. Independently decommission the R1 and R2 LUNs.
   a. Decommission the R1 LUN.

   ```
   FabricAdmin:switch> cryptocfg --decommission -container <container name>
   -initiator <initiator  PWWN> -LUN <lun number>
   ```

   b. Display the decommissioned key IDs.

   ```
   FabricAdmin:switch>cryptocfg --show –decommissionedkeyids
   ```

   c. Delete the respective key from the key vault. On the Brocade Encryption Switch, enter the following command.

   ```
   FabricAdmin:switch> cryptocfg --delete –decommissionedkeyids
   ```

   d. Decommission the R2 LUN.

   ```
   FabricAdmin:switch> cryptocfg --decommission -container <container name>
   -initiator <initiator  PWWN> -LUN <lun number>
   ```

Force-enabling a decommissioned disk LUN for encryption

When trying to re-use primary or secondary replicated LUNs, you must first decommission the LUNs. When trying to re-use a decommissioned LUN, you must:

1. Delete the keys from the key vault.
2. Log in as Admin or FabricAdmin.
3. Delete the decommissioned LUN IDs from the Brocade Encryption Switch.

    e. Display the decommissioned key IDs.

    ```
    FabricAdmin:switch> cryptocfg --show –decommissionedkeyids
    ```

    f. Delete the respective key from the Brocade Encryption Switch. Enter the following command.

    ```
    FabricAdmin:switch> cryptocfg --delete –decommissionedkeyids
    ```

4. Add the LUN back into the container as **cleartext.**

    ```
    FabricAdmin:switch> cryptocfg --add –LUN <crypto target container name> <LUN
    Num | LUN Num Range> <Initiator PWWN> <Initiator NWWN> -lunstate cleartext
    ```

5. Enable the LUN.

    ```
    FabricAdmin:switch> cryptocfg --enable -LUN <crypto target container name>
    <LUN Num> <Initiator PWWN>
    ```

6. Modify the LUN to encrypted.

    ```
    FabricAdmin:switch> cryptocfg --modify -LUN <crypto target container name>
    <LUN Num> <Initiator PWWN> 0 -lunstate encrypted -encryption_format native
    -encrypt
    ```

7. Enter the **cryptocfg --enable -LUN** command followed by the CryptoTarget container name, the LUN Number, and the initiator PWWN.

    ```
    FabricAdmin:switch> cryptocfg --enable -LUN my_disk_tgt 0x0 \
    10:00:00:00:c9:2b:c9:3a
    Operation Succeeded
    ```

Force-enabling a disabled disk LUN for encryption

You can force a disk LUN to become enabled for encryption when encryption is disabled on the LUN. A LUN may become disabled for various reasons, such as a change in policy from **encrypt** to **cleartext** when encrypted data (and metadata) exist on the LUN, a conflict between LUN policy and LUN state, or a missing DEK in the key vault. Force-enabling a LUN while metadata exist on the LUN may result in a loss of data and should be exercised with caution. Refer to Chapter 6, "LUN policy troubleshooting" on page 275 for a description of conditions under which a LUN may be disabled, and for recommendations on re-enabling the LUN while minimizing the risk of data loss.

This procedure must be performed on the local switch that is hosting the LUN. No commit is required to force-enable after executing this command.

1. Log in to the switch that hosts the LUN as Admin or FabricAdmin.
2. Enter the **cryptocfg --enable -LUN** command followed by the CryptoTarget container name, the LUN Number, and the initiator PWWN.

   ```
   FabricAdmin:switch> cryptocfg --enable -LUN my_disk_tgt 0x0 \
   10:00:00:00:c9:2b:c9:3a
   Operation Succeeded
   ```

SRDF LUNs

The Symmetrix Remote Data Facility (SRDF) transmits data that is being written to a local Symmetrix array to a remote symmetrix array. The replicated data facilitates a fast switch-over to the remote site for data recovery.

SRDF supports the following methods of data replication:

- Synchronous Replication provides real-time mirroring of data between the source Symmetrix and the target Symmetrix systems. Data is written simultaneously to the cache of both systems in real time before the application I/O is completed, thus ensuring the highest possible data availability.
- Semi-Synchronous Replication writes data to the source system, completes the I/O, and then synchronizes the data with the target system. Since the I/O is completed prior to synchronizing data with the target system, this method provides an added performance advantage. A second write will not be accepted on a Symmetrix source device until its target device has been synchronized.
- Adaptive Copy Replication transfers data from the source devices to the remote devices without waiting for an acknowledgment. This is especially useful when transferring large amounts of data during data center migrations, consolidations, and in data mobility environments.
- Asynchronous Replication places host writes into chunks and then transfers an entire chunk to the target system. When a complete chunk is received on the target system, the copy cycle is committed. If the SRDF links are lost during data transfer, any partial chunk is discarded, preserving consistency on the target system. This method provides a consistent point-in-time remote image that is not far behind the source system and results in minimal data loss if there is a disaster at the source site.

SRDF pairs

Remote replication is implemented by establishing a synchronized pair of SRDF devices connected by FC or IP links. A local source device is paired with a remote target device while data replication is occurring. While the SRDF devices are paired, the remote target device is not locally accessible for read or write operations. When the data replication operation is completed, the pair may be split to enable normal read/write access to both devices. The pair may be restored to restore the data on the local source device.

Figure 92 shows the placement of Brocade Encryption Switches in an SRDF configuration. When encryption is enabled for the R1 LUN, encrypted data written by the local application server to the R1 LUN is replicated on the R2 LUN. The data is encrypted using a DEK that was generated on the local encryption switch and stored on the local key vault. When each site has an independent key vault, the key vaults must be synchronized to ensure the availability of the DEK at the remote site. Both sites may share the same key vault, which eliminates the need for synchronization across sites. Depending on distance between sites, sharing a key vault might add some latency when retrieving a key.

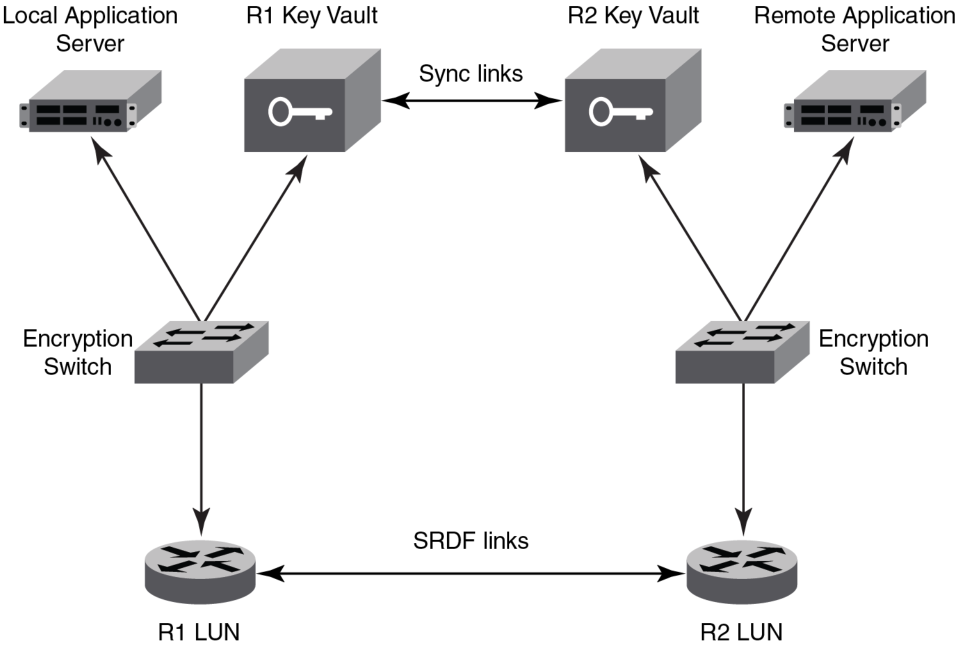


FIGURE 92 **Brocade Encryption Switches in an SRDF Configuration**

NOTE
When Symmetrix arrays are managed in-band, the gatekeeper LUNs must be added to the crypto-target containers as cleartext LUNs. Adding these as encrypted LUNs generates a CRITICAL error on the console, and the other encrypted LUNs are not visible from the host.

Enabling remote replication mode

To enable the remote replication features, issue the **cryptocfg --set -replication enable** command.The remote replication features are supported in Fabric OS 6.4 and later. Remote replication is disallowed under the following conditions:

- One of the nodes in an encryption group is currently running a Fabric OS version prior to v6.4.
- A node is downgraded to Fabric OS version prior to v6.4.

When replication mode is enabled, starting first-time encryption (FTE) or manual rekey on LUNs without metadata (due to uncompressible metadata blocks) generates a RASLOG entry, providing the key ID that is used to encrypt the LUN. Key expiry rekey (or auto rekey) is disabled for LUNs without metadata.

Replication mode can be disabled with the **cryptocfg --set -replication disable** command. This operation will fail if there are LUNs configured with the **-newLUN** option in the encryption group.

Once replication mode is enabled, the switch firmware cannot be downgraded to firmware versions prior to Fabric OS 6.4.0.

CAUTION

Do not add a node running an earlier Fabric OS version to an encryption group that is running version 6.4.0 or later if remote replication is enabled. Also, be aware that a Fabric OS 6.4.0 configuration file is not blocked from being downloaded to a node running an earlier Fabric OS version.

Adding replication LUNs

Replication LUNs must be added to the container with the -**newLUN** option. Replication mode needs to be enabled prior to adding replication LUNs with -**newLUN** option, using the **cryptocfg** --**set** -**replication enable** command. The primary LUN and all mirror LUNs need to be added to their respective containers with the -**newLUN** option.

From the standpoint of the encryption switch or blade, the local and remote copies of the LUN are configured in different encryption groups. From the DPM perspective, DPM clusters at local and remote encryption groups must be configured as part of the same DPM cluster group.

Rekey operations for replicated LUNs

Auto rekey is disabled for replicated LUNs. Sync between primary LUNs and mirror LUNs should be disabled before starting manual rekey on primary LUNs. If sync is not disabled, the mirror LUN will be disabled for host access. Once the primary LUN rekey is completed, the sync can be performed between the primary (R1) and mirror (R2) LUN. Manual rekey works only on primary LUNs. Mirror LUNs can be converted to primary LUNs by performing a manual rekey with the -**include_mirror** option.

Be aware that when an individual primary LUN is rekeyed using the -**include_mirror** option, no warning message is displayed prior to the rekey occurring.

If a rekey is invoked using the -**include_mirror** option, and the LUN is not a mirror LUN or a read-only primary LUN, the rekey operation acts as usual.

NOTE
cryptocfg --**manual_rekey** -**all** -**include_mirror** rekeys all the primary and mirror LUNs, not just mirror LUNs and out-of-sync primary LUNs. Enter only **cryptocfg** --**manual_rekey** -**all** if you want to rekey only out-of-sync primary LUNs. The -**include_mirror** option is ignored if the command applies only to a primary LUN.

Reading metadata after sync

The **cryptocfg** --**refreshDEK** command can be used to perform a read of the metadata and reprogram the encryption tables with a new encryption key. After a sync from rekeyed primary LUN to the mirror LUN, performing **cryptocfg** --**refreshDEK** will obtain the latest encryption keys for the primary LUN and configure that for encryption and decryption of the mirror LUN.

NOTE
For all multi-path LUN environments, it is critical to ensure that the target port settings (for example, os2007 bit, or spc-2 bit) for all paths to a given LUN are configured identically.

Using SRDF, TimeFinder and RecoverPoint with encryption

The EMC Symmetrix Remote Data Facility (SRDF), TimeFinder (TF), and RecoverPoint (RP) work together to provide reliable and efficient data recovery from a remote data facility:

- SRDF transmits data that is being written to a local Symmetrix array to a remote Symmetrix array. The replicated data facilitates a fast switchover to the remote site for data recovery.
- TF provides local storage replication for increased application availability and faster data recovery.
- RP facilitates continuous data protection and continuous remote replication to enable on-demand protection and point in time data recovery.

RecoverPoint Configuration Restrictions

The Brocade encryption solution only supports Clariion-based splitting operations. The Clariion arrays at both the source and target sites must be configured so that the RP appliances have dedicated cleartext target ports for access to the storage arrays. That is, every storage array target port utilized by a RP appliance for access to the Clariion array must not be configured as a CryptoTarget container.

Initial Configuration Requirements

The following are initial configuration requirements for SRDF, TF, and RP:

- For SRDF and RP, it is assumed that there is a clustered pair of DPMs at the local site and a clustered pair of DPMs at the remote site. The clustered pairs must then be configured as part of the same key vault group
- For TimeFinder, the source device (LUN) and the target device (LUN) must be located on the same storage array.

NOTE
If replication is enabled, firmware downgrades to earlier Fabric OS releases will be disallowed until the replication feature is disabled. The replication feature cannot be disabled if there are LUNs in the Encryption Group (EG) configured with the –newLUN option.

SRDF/RP initial setup at the source (R1) site

Replication mode needs to be enabled before replicated LUNs can be added to the Brocade Encryption Switch, and the master key must be exported.

1. Log in as Admin or SecurityAdmin.
2. Use the following command to enable EG wide replication mode:

```
SecurityAdmin:switch> cryptocfg --set -replication enable
```

3. Export the master key.

```
SecurityAdmin:switch> cryptocfg -exportmasterkey
```

4. Make a note of the master key's ID. The master key ID can be obtained by running the following command:

```
SecurityAdmin:switch> cryptocfg --show -localEE
```

NOTE
The master key is being exported from the local site so it can be recovered and utilized by the EG at the remote site. If the local and remote sites are both part of the same encryption group and therefore share the same DPM cluster, this step is not required.

SRDF/RecoverPoint remote target (R2) site

Replication mode needs to be enabled before replicated LUNs can be added to the Brocade Encryption Switch, and the master key configured on encryption group at the source (R1) site must be recovered for use on encryption group at the remote (R2) site.

1. Log in as Admin or SecurityAdmin.
2. Enable EG wide replication mode.

```
SecurityAdmin:switch> cryptocfg --set -replication enable
```

3. Recover the master key configured on the local site EG to the remote site EG.

```
SecurityAdmin:switch> cryptocfg --recovermasterkey currentMK -keyid <key ID of
master key from R1's EG>
```

   Recovery of the master key at the remote site needs to be accomplished before adding replicated LUNs to the encryption group configuration at the remote/target site.

Configuring LUNs for SRDF/TF or RP deployments

There are two possible LUN configuration scenarios LUNs to consider in SRDF/TF or RP deployments:

- Creating new source LUNs that can later be replicated.
- Migrating data from existing encrypted or cleartext source LUNs to LUNs that can be replicated.

For each of these scenarios, the following rules and notes apply:

- It is assumed that CryptoTarget containers (CTCs) have been created for all target ports at the local site (and at the remote site if one exists) and that the appropriate initiators have been added to each.
- SRDF R1 and R2 LUNs must be the same size.
- TimeFinder (TF) source and target devices (LUNs) must be the same size.
- RecoverPoint (RP) source and target devices (LUNs) must be the same size.
- When changing encryption policies for the source LUN, the same policies must be applied to the target LUN.
- Once the LUN is added to the container using the -newLUN option, it must not be resized.
- Auto/Key expiry rekey is not allowed for SRDF/TF/RP LUNs. Therefore the -newLUN option is not compatible with the -enable_rekey option.

Steps for dealing with these scenarios are described in the following sections devoted to using SRDF, TimeFinder (TF) and RecoverPoint (RP) with the Brocade encryption solution.

Creating new source LUNs that can later be replicated

Use the following command to create a new source LUN capable of later replication. This command must be completed once for every path/container that has access to the source LUN:

1. Log in as Admin or FabricAdmin.
2. Create the new source LUN with the –**newLUN** option and –**encrypt** policy

   ```
   FabricAdmin:switch> cryptocfg --add -LUN <source_container> <new LUN num>
   <initiator PWWN & NWWN> -newLUN -lunstate cleartext -encrypt
   ```

   NOTE
   This command assumes there is no valid user data on the LUN. Therefore, this command will have the effect of destroying any existing user data on the LUN.

3. Commit the configuration

   ```
   FabricAdmin:switch> cryptocfg --commit
   ```

Migrating LUNs with existing data to LUNs that can be replicated

As part of the encryption replication solution, if a SRDF/TF/RP source LUN contains valid customer data (cleartext or encrypted), prior to replicating the LUN, the existing user data must be migrated to a new LUN that is at least three blocks larger than the current source LUN.

If your setup has an existing target LUN, it too will need to be deleted and then recreated as a LUN that is identical in size to the new larger source LUN.

The steps for migrating data from the existing source LUN to a larger and replication-capable LUN depend on whether or not the existing LUN contains encrypted data or cleartext data. The two options are described below.

NOTE
R1 and R2 devices must be of the same size. If the LUNs are of different sizes and R1 does not have primary metadata, encryption of R2 LUNs will fail and the LUN becomes disabled. This is because the location of the secondary metadata differs for R1 and R2 LUNs that are not the same size. Normally, R1 writes secondary metadata at its last three blocks, but because R2 is a different size, its last three blocks do not contain metadata, causing the encryption setup to fail.

OPTION 1 (data migration for encrypted source LUNs)

1. Log in as Admin or FabricAdmin.
2. Create a new LUN with –**newLUN** option and –**encrypt** policy.

   ```
   FabricAdmin:switch> cryptocfg --add -LUN <source_container> <new LUN num>
   <initiator PWWN & NWWN> -newLUN -lunstate cleartext -encrypt
   ```

NOTE
All paths to the new SRDF/TF/RP source LUN must be added to their containers with the -**newLUN** option.

3. Commit the configuration.
4. Wait until the LUN is in **encryption enabled** state.
5. Copy the data from the old LUN to the new LUN using the EMC host-based PPME (PowerPath Migration Enabler) application. Information on PPME can be found on the EMC Powerlink website: http://powerlink.emc.com

OPTION 2 (data migration for cleartext source LUNs)

1. Log in as Admin or FabricAdmin.
2. Create a new LUN with -**newLUN** option and -**cleartext** policy.

```
FabricAdmin:switch> cryptocfg --add -LUN <source_container> <new LUN num>
<initiator PWWN & NWWN> -newLUN -lunstate cleartext -cleartext
```

NOTE
All paths to the new SRDF/TF/RP source LUN must be added to their containers with the -**newLUN** option.

3. Commit the configuration.
4. Copy the data from the old LUN to the new LUN using the EMC host-based EMC PPME (PowerPath Migration Enabler) application. Information on PPME can be found on the EMC Powerlink website: http://powerlink.emc.com
5. If first-time encryption of this LUN is required, configure the LUN for encryption and enable first-time encryption as follows:

```
FabricAdmin:switch> cryptocfg --modify -LUN <source_container> <new LUN num>
<initiator PWWN> -encrypt -enable_encexistingdata
```

NOTE
For multi-path LUNs, you must repeat this step for each path before committing the configuration.

6. Commit the configuration.

Synchronizing source and target LUN SRDF/RP pairs

This section describes the proper procedure for establishing the local/remote LUN pair in a SRDF or RP environment.

NOTE
The remote/target LUNs must be added to their CryptoTarget Containers (CTCs) only after the local site LUNs' encryption setup has been completed.

1. If necessary, create the remote/R2 LUN at the remote site ensuring that it is identical in size to the local/R1 site LUN. At this time, do not configure the remote LUN to be a part of any remote site CTC.
2. Establish the local-to-remote LUN replication/synchronization and wait for the pair to become fully synchronized.

   NOTE
   During the initial SRDF/RP replication (or while replicating/synchronizing after a rekey of the source LUN), the remote/R2 LUNs must not be exposed for access to the remote site hosts. Although the SRDF/RP behavior may make the remote/R2 LUN read-only or not-ready, it is mandated that the target ports be physically taken offline. Once synchronized, if remote access to the target LUN becomes necessary, the process of bringing the remote target ports online will ensure the correct Data Encryption Key (DEK) is injected into every Encryption Engine (EE) with a path to the remote LUN.

3. Verify the SRDF/RP pair is in a synchronized state using the EMC Solution Enabler or the RP GUI, depending on which technology you are implementing.
4. Verify that the DEKs are synchronized between the local and remote DPMs. This can be done manually for each LUN as follows:
   a. Issue the command **cryptocfg --show -vendorspecifickeyid key_ID** for each replicated LUN and capture the UUIDs (Universally Unique Identifier) returned.
   b. Search for this UUID on the remote key vaults to ensure its presence.

   Alternatively, simply bringing the remote site LUNs online ensures the remote DEKs are present. To bring the remote/R2 LUNs online, follow these steps:

   a. Bring all target ports through which the remote LUNs are accessible online.
   b. If not already created, add the remote/R2 CTCs for each path to each remote LUN.

   ```
   FabricAdmin:switch> cryptocfg --create -container disk <remote target
   container name> EE_node_WWN [EE_slot] target_PWWN target_NWWN [-initiator
   initiator_PWWN initiator_NWWN [initiator_PWWN initiator_NWWN]...]
   ```

   c. Add the remote/R2 LUNs to all of their respective CTCs.

   ```
   FabricAdmin:switch> cryptocfg --add -LUN <remote container name> <remote
   LUN ID> <initiator PWWN & NWWN> -lunstate encrypted -encrypt -newLUN
   ```

5. Commit the configuration.
6. Verify that the remote LUN states are “Encryption enabled” and their key IDs used for encryption are the same as those used by the local/R1 LUNs.

7. Verify that the Replication LUN type of the local/R1 LUNs is “Primary” and that of the remote/R1 LUNs is “Mirror.”
8. Take all remote target ports associated with CTCs through which the remote LUNs are accessible offline.

NOTE
If the DEK is not synchronized between the local/R1 site and the remote/R2 site, the remote/R2 LUN will automatically become disabled.

Configuring TimeFinder target devices

Use TimeFinder (TF) to create any desired target devices, such as snapshots, clones, or mirrors. The TimeFinder source and destination devices must be the same size.

NOTE
You may skip this section if host access to a target device is not required.

1. Log in as Admin or FabricAdmin.
2. Create the containers for the target ports through which TimeFinder target LUNs will be made accessible, and add all initiators that require access the target LUNs. If Target devices utilize the same target ports/containers as the source devices, you may skip this step.

```
FabricAdmin:switch> cryptocfg --create -container disk <target container name>
<EE to own container> <target PWWN> <target NWWN> -initiator <initiator PWWN>
<initiator NWWN>
```

3. For each path to the target device/LUN, add the LUNs to their respective target containers. All paths to the TimeFinder target LUN must be added to their containers with the –newLUN option.

```
FabricAdmin:switch> cryptocfg --add -LUN <target container name> <target LUN
ID> <initiator PWWN> <initiator NWWN> -newLUN -lunstate encrypted -encrypt
```

NOTE
If the target device specified above is a snapshot of a cleartext LUN, the above command results in that LUN becoming disabled. For cleartext snapshots, use the syntax –lunstate cleartext –cleartext.

4. Commit the configuration.
5. Verify the target LUN state shows “encryption enabled” and the key ID used for encryption is the same as the source LUN's key ID.

Configuring SRDF Gatekeeper LUNs

Gatekeeper LUNs used by SYMAPI on the host for configuring SRDF using in-band management must be added to their containers with a LUN state of **cleartext**, encryption policy of **cleartext**, and without the -**newLUN** option.

SRDF/TF/RP manual rekeying procedures

The following topics describe encryption rekeying procedures relative to SRDF, TF, and RP.

TF snapshot rekeying details

In TimeFinder environments, rekeying a source LUN which has one or more snapshot target devices will result in full copy outs of the source devices to the target devices.

When source LUNs are rekeyed, the target snapshot LUNs will continue to utilize the older/original DEK and therefore use of the refreshDEK command is not required. However, if an existing target LUN/snapshot is recreated, then the **refreshDEK** command must be used on every path/container which has access to the target device. The **refreshDEK** command forces the Brocade Encryption Switch to re-read the metadata on the target LUN and then updates the FPGA tables for the LUN if the DEK in the metadata has changed.

```
FabricAdmin:switch> cryptocfg --refreshDEK <target_container> <target LUN ID>
<initiator PWWN>
```

NOTE
Manual rekeying is supported for TimeFinder snapshot target device LUNs using the -**include_mirror** option; however, it would defeat the purpose of using snapshot LUNs because rekeying them would cause all blocks of the snapshot to be allocated to the virtual device (i.e. the source and snap LUNs would have the same number of blocks).

TF clone/mirror rekeying details

Manual rekeying is supported for TimeFinder source LUNs and is not supported for target devices (clone, mirror) unless the source to target connection is first split.

1. Log in as Admin or Fabric Admin.
2. Split the TF source/target LUN pair ensuring the data synchronization from the source LUN to the destination LUN has been stopped.

   NOTE
   During all rekeying operations, data synchronization between the source and target LUN must be stopped.

3. During the rekeying operation, if desired, you can enable the target ports so the target LUNs can be accessed by the hosts in read-only mode.
4. Issue a manual rekey request for the source LUN.

   ```
   FabricAdmin:switch> cryptocfg --manual_rekey <source container> <source LUN
   ID> <initiator PWWN>
   ```

5. Wait until the rekey operation on the source LUN has completed. If the source LUN has a rekeying error of any type, the TF source/target LUN pair should not be established/synchronized. The source LUN rekey must complete successfully before the source/target pair is re-established.

6. Remove target LUN access by using one of the following procedures:

   - Ensure that no hosts read or write information to the TF target LUNs, or
   - Make the target LUN not ready to the host by using the –**not_ready** option of TF clone/snap when activating the target device.

7. Start TimeFinder pair synchronization so the rekeyed data from the source LUN is copied to target LUN.

8. Verify that the TimeFinder pair is synchronized.

9. If you want to bring the target LUN online for host access, once the TimeFinder pair has been synchronized, perform the following command on every path/container that has access to the target device:

```
FabricAdmin:switch> cryptocfg --refreshDEK <target_container> <target LUN ID>
<initiator PWWN>
```

NOTE
The refreshDEK command forces the Brocade Encryption Switch to re-read the metadata on the target LUN, and then updates the FPGA tables for the LUN if the DEK in the metadata has changed. It is therefore imperative that this command be run after every rekeying operation that is completed for TF target devices.

Rekeying local site (R1) SRDF LUNs

Manual rekeying is supported for SRDF R1 LUNs. If it is required to rekey the R2 LUN, SRDF role reversal/swap is necessary. This procedure is covered in "Rekeying remote site (R2) SRDF LUNs".

1. Log in as Admin or FabricAdmin.

2. Split the SRDF R1/R2 LUN pair ensuring that the data replication from the source R1 LUN to the destination R2 LUN has been stopped.

NOTE
During all rekeying operations, data replication between the source and target LUN must be stopped.

3. During the rekeying operation, if desired, you can enable the remote target ports so the target LUNs can be accessed by the remote hosts in read-only mode.

4. Issue a manual rekey request for the source LUN.

```
FabricAdmin:switch> cryptocfg --manual_rekey <R1 source container> <R1 source
LUN ID> <initiator PWWN>
```

5. Wait until the rekey operation on the source LUN has completed. If the source LUN has a rekeying error of any type, the SRDF pair should not be established/synchronized. The source LUN rekey must complete successfully before the source/target pair is re-established.

6. After confirming that the rekey has completed on the source LUN, perform the following to re-establish the source-to-target LUN replication:

   a. Remove the target LUN access by disabling all remote site target ports with access to the target LUN.

      NOTE
      In environments in which the target ports through which the target LUNs are accessible cannot be taken offline because they are used to access other LUNs, before remote access to the R2 LUNs is established, the refreshDEK command must be issued for all CTCs associated with the remote LUNs after the source LUNs have been rekeyed and synchronized with their target LUNs.

   b. Re-establish the SRDF R1/R2 LUN pair so that the rekeyed data from the source LUN is copied to the target LUN.

   c. Verify that the SRDF pair is in a fully synchronized state using the EMC Solution Enabler.

   d. Verify that the DEKs are synchronized between the local and remote DPMs. This can be done manually for each LUN as follows:

      1. Issue the **cryptocfg --show -vendorspecifickeyid key_ID** command for each replicated LUN and capture the UUIDs (Universally Unique Identifier) returned.

      2. Search for this UUID on the remote DPMs to ensure its presence.

      Alternatively, simply bringing the remote site LUNs online to the remote EEs ensures that the remote DEKs are present. To bring the remote LUNs online use following steps:

      1. Restore target LUN access by enabling all remote site target ports (associated with remote site CTCs) with access to the target LUN.

      2. Verify that the remote LUN states are **encryption enabled** and their key IDs used for encryption are the same as those used by the local site LUNs.

      3. Take all target ports associated with CTCs through which the remote LUNs are accessible offline.

      NOTE
      If the DEK is not synchronized between the local and remote sites, the remote LUN will automatically become disabled.

Rekeying LUNs for RP deployments - local site

Manual rekeying is supported for RP source LUNs. If it is required to rekey the remote LUN, RP role reversal/swap is necessary as described in “Rekeying LUNs for RP deployments - remote site”.

1. Log in as Admin or FabricAdmin.

2. Disable the RP source/target LUN consistency group, ensuring that the data replication from the source LUN to the destination LUN has been stopped.

   NOTE
   During all rekeying operations, data replication between the source and target LUN must be stopped.

3. During the rekeying operation, if desired, you can enable the remote targets ports so the target LUNs can be accessed by the remote hosts in read-only mode.
4. Issue a manual rekey request for the source LUN.

   ```
   FabricAdmin:switch> cryptocfg --manual_rekey <source container> <source LUN
   ID> <initiator PWWN>
   ```

5. Wait until the rekey operation on the source LUN has completed. If the source LUN has a rekeying error of any type, the RP pair consistency group should not be enabled. The source LUN rekey must complete successfully before the source/target pair consistency group gets re-enabled. After confirming that the rekey has completed on the source LUN, complete the following steps to re-establish the source to target LUN replication.

   a. Remove target LUN access by disabling all remote site target ports with access to the target LUN.

      NOTE
      In environments in which the target ports through which the target LUNs are accessible cannot be taken offline because they are used to access other LUNs, before remote access to the remote LUNs is established, the refreshDEK command must be issued for all CTCs associated with the remote LUNs after the source LUNs have been rekeyed and synchronized with their target LUNs.

   b. Enable the source/target LUN consistency group so that the rekeyed data from the source LUN is copied to target LUN.

   c. Verify that the RP pair is fully synchronized state using the RP GUI.

   d. Verify that the DEKs are synchronized between the local and remote DPMs. This can be done manually for each LUN as follows:

      1. Issue the command **cryptocfg --show -vendorspecifickeyid key_ID** for each replicated LUN and capture the UUIDs (Universally Unique Identifier) returned
      2. Search for this UUID on the remote DPMs to ensure its presence.

      Alternatively, simply bringing the remote site LUNs online to the remote EEs ensures the remote DEKs are present. To bring the remote LUNs online use following steps:

      1. Restore target LUN access by enabling all remote site target ports (associated with remote site CTCs) with access to the target LUN.
      2. Verify that the remote LUN states are **encryption enabled** and their key IDs used for encryption are the same as those used by the local site LUNs.
      3. Take all target ports associated with CTCs through which the remote LUNs are accessible offline.

After the rekey has completed, restoring from a bookmark taken prior to the rekey operation will result in the source LUN becoming READ ONLY. Once you have restored from the bookmark, it is imperative that you issue the **refreshDEK** command on all paths with access to the restored LUN.

NOTE
If the DEK is not synchronized between the local and remote sites, the remote LUN will automatically become disabled.

Rekeying remote site (R2) SRDF LUNs

To rekey an R2 LUN, you must first do an SRDF role reversal. Complete the following steps to reverse the R1/R2 LUN functional roles:

1. Issue the SRDF role swap command to change the old R1 LUN to the new R2 LUN and old R2 LUN to the new R1 LUN.
2. Split the SRDF pair.
3. Issue the **cryptocfg --manual_rekey <crypto target container name> <LUN Num> <Initiator PWWN> -include_mirror** command on the new R1 LUN (old R2 LUN).

   NOTE
   This command will fail with an error if the **-include_mirror** option is not provided with the **manual_rekey** request.

4. After the rekey is completed, disable the new R2 target ports.
5. Establish the SRDF for replication and wait for the SRDF pair to be fully synchronized.
6. Verify that the DEKs are synched up from the local site key vault cluster to the remote site key vault cluster.

   NOTE
   In all operations prior to SRDF establishment, ensure that the DEKs are synchronized between the local and remote site key vaults.

7. Verify that the Replication LUN type of the new R1 LUN is now “Primary” and the Replication LUN type of new R2 LUN is now “Mirror”.

   NOTE
   Verify the DEKs and Replication LUN type for all multi-paths are consistent.

Rekeying LUNs for RP deployments - remote site

To rekey a remote site LUN, you must first do an RP reverse direction. Complete the following steps to reverse the local LUN and remote LUN RP functional roles:

1. Issue the RP reverse direction command to change the old local LUN to the new remote LUN and old remote LUN to the new local LUN.
2. Disable the RP source/target LUN consistency group
3. Issue the **cryptocfg --manual_rekey -include_mirror <new local LUN container> < new local LUN ID> <initiator PWWN>** command on the new local LUN (old remote LUN).

   NOTE
   This CLI command will fail with an error if the **-include_mirror** option is not provided with the **manual_rekey** request

4. After the rekey is completed, disable the new remote target ports.
5. Enable the RP source/target LUN consistency group and wait for the RP pair to be fully synchronized.

6. Verify that the DEKs are synched up from local site DPM cluster to the remote site DPM cluster.

NOTE
In all operations prior to enabling the RP source/target LUN consistency group, ensure that the DEKs are synchronized between the local and remote site key vaults.

Behavior with Hosts writing beyond reported capacity

If a host writes beyond the reported capacity of a source or destination LUN, it can cause the LUN to become disabled when exposed. Hosts must honor the READ CAPACITY10/READ CAPACITY16 data returned by the Brocade Encryption Switch for SRDF/TF/RP source and destination LUNs.

Tape pool configuration

Tape pools are used by tape backup application programs to group all configured tape volumes into a single backup to facilitate their management within a centralized backup plan. A tape pool is identified by either a name or a number, depending on the backup application. Tape pools have the following properties:

- They are configured and managed per encryption group at the group leader level.
- All encryption engines in the encryption group share the same tape pool policy definitions.
- Tape pool definitions are only used when writing tapes. The tape contains enough information (encryption method and key ID) to enable any encryption engine to read the tape.
- Tape pool names and numbers must be unique within the encryption group.
- If a given tape volume belongs to a tape pool, tape pool-level policies (defaults or configured values) are applied and override any LUN-level policies.
- Tape drive (LUN) policies are used if no tape pools are created or if a given tape volume does not belong to any configured tape pools.

NOTE
Tape pool configurations must be committed to take effect. Expect a five second delay before the commit operation takes effect.There is an upper limit of 25 on the number of tape pools you can add or modify in a single commit operation. Attempts to commit a configuration that exceeds this maximum fails with a warning.

Tape pool labeling

Tape pools may be identified by either a name or a number depending on your backup application. Numbers are always entered and displayed in hex notation. Names and numbers are independent; it is possible to have one tape pool with the name ABC and another with the hex number ABC.

The following rules apply when creating a tape pool label:

- Tape pool names are limited in length to 63 characters. They may contain alphanumeric characters, and in some cases, underscores (_) and dashes (-).
- Tape pool numbers are limited to eight hex digits. Valid characters for tape pool numbers are 0-9, A-F, and a-f.

- The tape pool label created on the encryption switch or blade *must be the be same* tape pool label configured on the tape backup application.
- Refer to the tape backup product documentation for detailed instructions for creating tape pool labels and numbers.

  NOTE
  It may be helpful to create the tape pool on the application first to determine possible naming restrictions, then use the label generated by the backup application to create the tape pool on the encryption switch or blade.

- A tape pool must first be created on the encryption switch or blade before you can label the tape media and assign them to the tape pool. Failure to observe this sequence invalidates tape pool-level settings and policies, and default LUN-level settings are applied to the tape media.

CommVault Galaxy labeling

CommVault uses a storage policy for each backup. When configuring a tape pool to work with CommVault Galaxy, first create a storage policy on CommVault and then use the *storage_policy_id (sp_id)* as the label when creating the tape pool on the encryption switch or blade.

1. Open CommCellExplorer Views by selecting **Start** > **Programs** >**Microsoft SQL Server 2005** >**SQL ServerManagement Studio.**
2. Expand the tree in the left pane and navigate to the following location: *Comm_serve_computer_name\database_instance_name* >**Databases** > **CommServ** >**Views.**
3. Edit the **dbo.CommCellStoragePolicyquery** as follows:
   a. Right-click the view and select **Edit.**
   b. Add the following (sp_id= ARG.id) as follows:
      - SELECT Distinct
      - storagepolicy= ARG.name,
      - sp_id= ARG.id,
4. Save the query by selecting **File** > **Save SQLQuery1.sql**
5. Execute the query by right-clicking the query window and selecting **Execute.**
6. Open the **dbo.CommCellStoragePolicy** view.
7. Right-click the view **dbo.CommCellStoragePolicy** and select **Open View.**
8. Use the *sp_id* for the storage policy as the tape pool label on the encryption switch or blade.

NetBackup labeling

NetBackup uses numbers to label tape pools. If you are using NetBackup as your application, follow these steps to obtain the tape pool number.

1. Log into the NetBackup application Windows host.
2. Select **Start > run**, and type **cmd** in the dialog box.
3. Navigate to C:\Program Files\VERITAS\Volmgr\bin and enter the following command:

```
C:\Program Files\VERITAS\Volmgr\bin>vmpool -listall
```

```
============================================================
pool number:  0
pool name:    None
description:  the None pool
pool host:    ANYHOST
pool user:    ANY
pool group:   NONE
============================================================
```

4. Use the pool number as the tape pool number on the encryption switch or blade.

NetWorker labeling

NetWorker does not allow underscore characters in tape pool names. To ensure that you can use the same tape pool name on your encryption platform and on your backup application, create the tape pool on NetWorker first before creating the tape pool on your encryption switch.

Creating a tape pool

Complete the following steps to create a tape pool:

1. Log in to the group leader as FabricAdmin.
2. Create a tape pool by entering the **cryptocfg --create -tapepool** command. Provide a label or numeric ID for the tape pool and specify the encryption policies. For policies not specified at this time, LUN-level settings apply.
   - Set the tape pool policy to either **encrypt** or **cleartext** (default).
   - Set the encryption format to Brocade **native** (default)
   - Optionally set an expiration date in days for the key (default is no expiration). If the **key_lifespan** parameter is set at the tape pool level to a value other than none (default), the tape value is used over any LUN-level settings. If the **key_lifespan** parameter is not set at the tape level (default of none), LUN level settings apply.

   The following example creates a tape pool named “my_tapepool”.

   ```
   FabricAdmin:switch> cryptocfg --create -tapepool -label my_tapepool
   Operation succeeded.
   ```

3. Commit the transaction.

   ```
   FabricAdmin:switch> cryptocfg --commit
   Operation succeeded.
   ```

4. Display the configuration. Enter the **cryptocfg --show -tapepool** command followed by the tape pool number or label and the **-cfg** parameter.

   ```
   FabricAdmin:switch> cryptocfg --show -tapepool -label my_tapepool -stat
   Number of tapepool session(s):  1
   Tapepool 1:
   Tapepool label:          my_tapepool
   Encryption mode:         encrypted
   Encryption format:        native
   Number of sessions:       0
   Tape sessions within the pool:
   Operation succeeded.
   ```

5. Configure the tape pool on your backup application with the same tape pool label you used to create the tape pool on the encryption switch or blade. Refer to the manufacturer's product documentation for instructions.
6. On your backup application, label the tape media to assign to the tape pool. Refer to the manufacturer's product documentation for instructions.

Deleting a tape pool

This command does not issue a warning if the tape pool being deleted has tape media or volumes that are currently accessed by the host. Be sure the tape media is not currently in use.

1. Log in to the group leader as FabricAdmin.
2. Enter the **cryptocfg --delete -tapepool** command followed by a tape pool label or number. Use **cryptocfg --show -tapepool -all** to display all configured tape pool names and numbers.

```
FabricAdmin:switch> cryptocfg --delete -tapepool -label my_tapepool
Operation succeeded.
```

3. Commit the transaction

```
FabricAdmin:switch> cryptocfg --commit
Operation succeeded.
```

Modifying a tape pool

1. Log in to the group leader as Admin or FabricAdmin.
2. Enter the **cryptocfg --modify -tapepool** command followed by a tape pool label or number. Then specify a new policy, encryption format, or both. The following example changes the encryption format from Brocade native to DF-compatible.

```
FabricAdmin:switch> cryptocfg --modify -tapepool -label my_tapepool
-encryption_format DF_compatible
Operation succeeded.
```

3. Commit the transaction.

```
FabricAdmin:switch> cryptocfg --commit
Operation succeeded.
```

Impact of tape pool configuration changes

Tape pool-level policies overrule policy configurations at the LUN level, when no policies are configured at the tape pool level. The following restrictions apply when modifying tape pool-level configuration parameters:

- If you change the tape pool policy from **encrypt** to **cleartext** or from **cleartext** to **encrypt** while data is written to or read from a tape backup device, the policy change is not enforced until the current process completes and the tape is unmounted, rewound, or overwritten. This mechanism prevents the mixing of cleartext data to cipher-text data on the tape.
- You cannot modify the tape pool label or the key lifespan value. If you wish to modify these tape pool attributes, delete the tape pool and create a new tape pool with a different label and key lifespan.

Configuring a multi-path Crypto LUN

A single LUN may be accessed over multiple paths. A multi-path LUN is exposed and configured on multiple CryptoTarget Containers located on the same encryption switch or blade or on different encryption switches or blades.

CAUTION

When configuring a LUN with multiple paths, there is a considerable risk of ending up with potentially catastrophic scenarios where different policies exist for each path of the LUN, or a situation where one path ends up being exposed through the encryption switch and other path has direct access to the device from a host outside the secured realm of the encryption platform. Failure to follow proper configuration procedures for multi-path LUNs results in data corruption.

To avoid the risk of data corruption, you *must* observe the following rules when configuring multi-path LUNs:

- During the initiator-target zoning phase, complete in sequence all zoning for ALL hosts that should gain access to the targets before committing the zoning configuration.
- Complete the CryptoTarget container configuration for ALL target ports in sequence and add the hosts that should gain access to these ports *before* committing the container configuration. Upon commit, the hosts lose access to all LUNs until the LUNs are explicitly added to the CryptoTarget containers.
- When configuring the LUNs, the *same* LUN policies must be configured for ALL paths of ALL LUNs. Failure to configure all LUN paths with the same LUN policies results in data corruption.

Multi-path LUN configuration example

Figure 93 on page 199 shows a single LUN on a dual-port target that is accessed over two paths by a dual-port host. The two encryption switches form an encryption group and an HA cluster. The following example illustrates a simplified version of a multi-path LUN configuration.

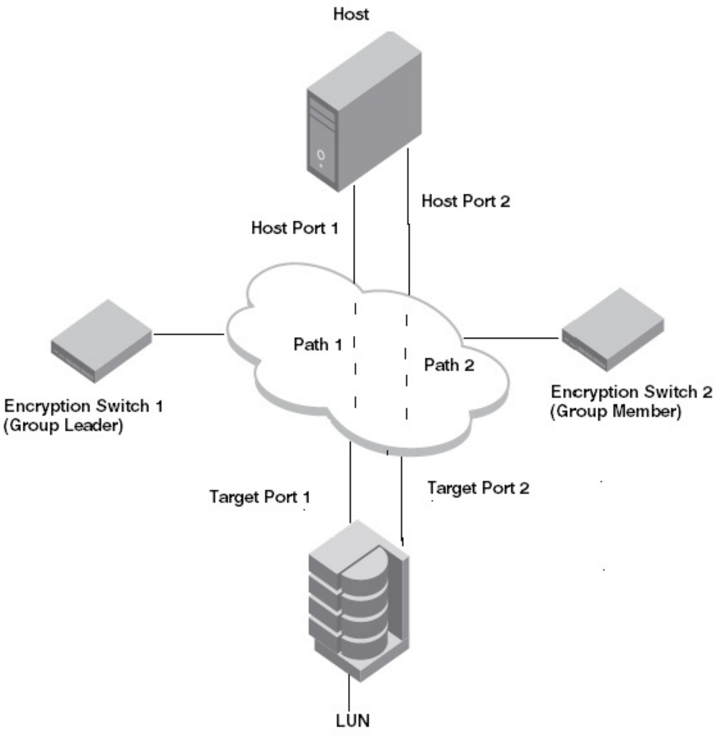


FIGURE 93 A LUN accessible through multiple paths

The following steps may be used to configure multiple path access to the LUN in Figure 93.

1. Create zoning between host port 1 and target port 1. Refer to the section "Creating an initiator - target zone" on page 158 for instructions.
2. Create zoning between host port 2 and target port 2. Refer to the section "Creating an initiator - target zone" on page 158 for instructions.
3. On the group leader encryption switch (switch 1), create a CryptoTarget container for each target port and add the hosts in sequence. Do NOT commit the configuration until you have created all CryptoTarget containers and added all hosts to the respective containers.
    a. Create a CryptoTarget container (CTC1) for target port 1 to be hosted on the encryption engine of encryption switch 1. Refer to the section "Creating a CryptoTarget container" on page 162 for instructions on steps b. through e.

```
FabricAdmin:switch> cryptocfg --create -container disk CTC1 \
<switch 1 WWN> 0 <Target Port 1 WWN> <Target NWWN>
```

b. Create a CryptoTarget container (CTC2) for target port 2 to be hosted on the encryption engine of encryption switch 2.

```
FabricAdmin:switch> cryptocfg --create -container disk  CTC2 \
<switch 2 WWN> 0 <Target Port2 WWN> <Target NWWN>
```

c. Add host port 1 to the container CTC1.

```
FabricAdmin:switch> cryptocfg --add -initiator <CTC1> <Host Port1 WWN> \
<Host NWWN>
```

d. Add host port 2 to the container CTC2.

```
FabricAdmin:switch> cryptocfg --add -initiator <CTC2> <Host Port2 WWN>
<Host NWWN>
```

e. Commit the configuration.

```
FabricAdmin:switch> cryptocfg --commit
```

Upon commit, redirection zones are created for target port 1, host port 1 and target port 2, host port 2. These redirection zones include the virtual target VT1 for CTC1, the virtual initiator VI1 for host port 1, the virtual target VT2 for CTC2 and the virtual initiator VI2 for host port 2. At this stage, the host loses access to all LUNs until the LUNs are explicitly added to the CryptoTarget containers.

4. Discover the LUNs. Perform steps 4 a. through c. to discover the LUNs for ALL CryptoTarget containers in sequence. Refer to the section "Discovering a LUN" on page 167 for details on the LUN discovery process and a command output example.

   a. On the encryption switch 1 (the group leader), enter the **cryptocfg --discoverLUN** for the container CTC1. The command output displays the LUNs present in the target as exposed from target port 1 and as seen by host port1, the LUN Number, host port1 WWN, and the LUN Serial Number.

   ```
   FabricAdmin:switch> cryptocfg --discoverLUN CTC1
   ```

   b. On the encryption switch 2, enter the **cryptocfg --discoverLUN** for the container CTC2. The command output displays the LUNs present in the target as exposed from target port and as seen by host port 2, the LUN Number, host port1 WWN, and the LUN Serial Number.

   ```
   FabricAdmin:switch> cryptocfg --discoverLUN CTC2
   ```

   c. Review the output of the LUN discovery to ensure that the LUN serial number for ALL LUNs are the same as seen from target-port 1 to host-Port 1 path and from target-port 2 to host-port 2. Identical LUN serial numbers validate the multi-path configuration.

5. Configure the LUN for all CryptoTarget containers in sequence by adding the LUN to each CryptoTarget container with identical policy settings. Refer to the sections "Configuring a Crypto LUN" on page 168 and "Crypto LUN parameters and policies" on page 169 for more information.

   a. Add the LUN to the CryptoTarget container CTC1 with policies.

   ```
   FabricAdmin:switch> cryptocfg --add -LUN CTC1 0 <Host Port1 WWN> \
   <Host NWWN> -lunstate cleartext -encryption_format native -encrypt \
   -enable_encexistingdata -enable_rekey 10
   ```

b. Add the same LUN to the CryptoTarget container CTC2. Use exactly the same LUN state and policy settings that you used for the LUN added to CTC1.

```
FabricAdmin:switch> cryptocfg --add -LUN CTC2 0 <Host Port1 WWN> \
<Host NWWN> -lunstate cleartext -encryption_format native -encrypt \
-enable_encexistingdata -enable_rekey 10
```

NOTE

The LUN policies must be exactly the same on both CTC1 and CTC2. Failure to do so results in undefined behavior and data corruption.

6. Validate the LUN policies for all containers. Display the LUN configuration for ALL CryptoTarget containers to confirm that the LUN policy settings are the same for all CryptoTarget containers.

```
FabricAdmin:switch> cryptocfg --show -LUN CTC1 0 <Host Port1 WWN> -cfg
FabricAdmin:switch> cryptocfg --show -LUN CTC2 0 <Host Port2 WWN> -cfg
```

Example:

```
FabricAdmin:switch> cryptocfg --show -LUN cx320-157A 0x1
10:00:00:00:c9:56:e4:7b -cfg
EE node:                10:00:00:05:1e:40:4c:00
EE slot:                9
Target:                 50:06:01:60:30:20:db:34 50:06:01:60:b0:20:db:34
VT:                     20:00:00:05:1e:53:8d:cd 20:01:00:05:1e:53:8d:cd
Number of host(s):      1
Configuration status:   committed
Host:                   10:00:00:00:c9:56:e4:7b 20:00:00:00:c9:56:e4:7b
VI:                     20:02:00:05:1e:53:8d:cd 20:03:00:05:1e:53:8d:cd
LUN number:             0x1
LUN type:               disk
LUN CFG state:          encrypted
Encryption mode:        encrypt
Encryption format:      native
Encrypt existing data:  disabled
Rekey:                  enabled
Key ID:                 not available
New LUN:                No
Key life:       30 (days) 0 (minutes)
Operation succeeded.
```

7. Commit the LUN configuration.

```
FabricAdmin:switch> cryptocfg --commit
```

Make sure the LUNs in previously committed LUN configurations and LUN modifications have a LUN state of Encryption Enabled before creating and committing another batch of LUN configurations or modifications.

NOTE

A maximum of 25 disk LUNs can be added or modified in a single commit operation. The maximum commit for tape LUNs is eight. Attempts to commit configurations or modifications that exceed the maximum commit allowed will fail with a warning. There is a five-second delay before the commit operation takes effect.

First-time encryption

First-time encryption, also referred to as encryption of existing data, is similar to the rekeying process described in the previous section, except that there is no expired key and the data present in the LUN is cleartext to begin with.

In a first-time encryption operation, cleartext data is read from a LUN, encrypted with the current key, and written back to the same LUN at the same logical block address (LBA) location. This process effectively encrypts the LUN and is referred to as “in-place encryption.”

Resource allocation

System resources for first-time encryption sessions are shared with rekey sessions. There is an upper limit of 10 sessions with two concurrent sessions per target. Refer to the rekey “Resource allocation” on page 202 section for details.

First-time encryption modes

First-time encryption can be performed under the following conditions:

- **Offline encryption:** The hosts accessing the LUN are offline or host I/O is halted while encryption is in process.
- **Online encryption:** The hosts accessing the LUN are online and host I/O is active during the encryption operation.

Configuring a LUN for first-time encryption

First-time encryption options are configured at the LUN level either during LUN configuration with the **cryptocfg --add -LUN** command, or at a later time with the **cryptocfg --modify -LUN** command.

1. Set the LUN policy to **encrypt** to enable encryption on the LUN. All other options related to encryption are enabled. A DEK is generated and associated with the LUN.
2. Enable first-time encryption by setting the **-enable_encexistingdata** parameter. The existing data on the disk is encrypted using the configured DEK.
3. Optionally set the auto rekeying feature with the **cryptocfg -enable_rekey** command and specify the interval at which the key expires and automatic rekeying should take place (*time period in days*) Enabling automatic rekeying is valid only if the LUN policy is set to **encrypt** and the encryption format is Brocade **native.** Refer to the section “Crypto LUN parameters and policies” on page 169 for more information.

The following example configures a LUN for first-time encryption with rekeying scheduled at a 6-month interval. You must commit the operation to take effect.

```
FabricAdmin:switch> cryptocfg --add -LUN my_disk_tgt 0x0 \
10:00:00:00:c9:2b:c9:3a 20:00:00:00:c9:2b:c9:3a -encrypt \
-enable_encexistingdata -enable_rekey 180
Operation Succeeded
```

Thin provisioned LUNs

With the introduction of Fabric OS 7.1.0, the Brocade Encryption Switch can discover if a disk LUN is thin provisioned LUN. Support for a thin provisioned LUN is limited to disk containers only.

NOTE
Currently, thin provisioned LUN support is limited to Brocade-tested storage arrays. The thin provisioned LUN status will be displayed as Yes for supported storage arrays running specific supported firmware versions only. Contact your service representative to determine if your storage array is supported.

Thin provisioned LUNs rely on on-demand allocation of blocks of data, instead of the traditional method of allocating all blocks up front. If a thin provisioned LUN status is shown as Yes, then first-time encryption and rekey are done on the allocated blocks only, which results in the provisioned region of the LUN remaining the same after the rekey is performed.

Thin provisioned LUN support requires no action by the user; the Brocade Encryption Switch can automatically detect if a LUN is a thin provisioned LUN. You can, however, identify a thin provisioned LUN using the following commands.

cryptocfg --show –container –all –stat

cryptocfg --discoverLUN -container

cryptocfg --show -rekey –all

cryptocfg --show –LUN tpdisk 0 10:00:00:00:c9:29:0f:01 –stat

NOTE:

- If a LUN is a thin provisioned LUN, TP LUN status is shown as Yes. (Thin provision support is limited to Brocade-tested storage arrays. The thin provisioned LUN status will be displayed as Yes for supported storage arrays only.)
- If a LUN is not a thin provisioned LUN or if thin provisioning is not supported with the LUN, LUN status is shown as No. (This can be a result of the array not supporting thin provisioning, or the Brocade Encryption Switch/blade does not support the thin provisioning features of the array. Refer to the Fabric OS release notes for supported arrays.)
- If LUN status cannot be determined, LUN status is shown as Unknown.

```
FabricAdmin:switch> cryptocfg --show -container -all -stat
    LUN number:              0xd
    LUN type:                disk
    LUN serial number:       50002AC000BC0A50
    Encryption mode:         encrypt
    Encryption format:       native
    Encrypt existing data:   disabled
    Rekey:                   disabled
    Internal EE LUN state:   Encryption enabled
    Encryption algorithm:    AES256-XTS
    Key ID state:            Read write
    New LUN:                 No
    TP LUN: Yes
    Key ID:                  4b:d9:4d:12:93:67:0e:0d:d1:e0:ca:aa:ba:34:29:db
    Key creation time:       Thu Sep 15 18:01:01 2011

FabricAdmin:switch> cryptocfg --discoverLUN -container
    Host:                    21:00:00:e0:8b:90:7c:c0
    LUN number:              0xd
    LUN serial number:      50002AC000BC0A50
    TP LUN:              Yes
    LUN connectivity state: Connected
    Key ID state:           Key ID not Applicable

FabricAdmin:switch> cryptocfg --show -rekey –all
    LUN number:             0x0
    LUN serial number:      50002AC002E70A50
    TP LUN:Yes
    Rekey session number:   0
    Percentage complete:    98
    Rekey state:            Read Phase
    Rekey role:             Primary/Active
    Block size:             512
    Number of blocks:       4194304
    Current LBA:            4141617

FabricAdmin:switch> cryptocfg --show –LUN tpdisk 0 10:00:00:00:c9:29:0f:01 –
stat
    LUN number:             0x0
    LUN type:               disk
    LUN serial number:      50002AC002E70A50
    TP LUN:Yes
    Encryption mode:        encrypt
    Encryption format:      native
    Encrypt existing data:  disabled
    Rekey:                  disabled
    Internal EE LUN state:  Encryption enabled
    Encryption algorithm:   AES256-XTS
    Key ID state:           Read write
    New LUN:                No
    Key ID:                 d3:a6:de:b9:67:26:04:fa:b9:83:e1:c7:cf:ae:f1:9c
    Key creation time:      Fri Feb 17 03:41:54 2012
```

Space reclamation

When a block that was provisioned is no longer needed, it can be reclaimed. The Brocade Encryption Switch supports following methods to reclaim the provisioned blocks:

- Sending the UNMAP SCSI command
- Sending WRITE_SAME SCSI command with only UNMAP bit set.

Note the following limitations:

- Space reclamation is not allowed during rekey.
- The Host will get garbled data while trying to read an unmapped region.
- Because windows host utility “sdelete –c” sends WRITE command with zeros to unmap LBAs, and which is currently not supported on the Brocade Encryption Switch, this utility will not be able to unmap LBAs.
- Rekey temporarily uses the last 512 blocks. As a result, these blocks will be marked as provisioned by the thin provisioned LUN.
- The first 16 blocks of the LUN will be mapped automatically (if it was unmapped), after the LUN has been configured as an encrypted LUN.

Data rekeying

In a rekeying operation, encrypted data on a LUN is decrypted with the current key, re-encrypted with a new key and written back to the same LUN at the same logical block address (LBA) location. This process effectively re-encrypts the LUN and is referred to as “in-place rekeying.”

It is recommended that you limit the practice of rekeying to the following situations:

- Key compromise as a result of a security breach.
- As a general security policy to be implemented as infrequently as every six months or once per year.

Rekeying is only applicable to disk array LUNs or fixed block devices. There is no rekeying support for tape media. If there is a need to re-encrypt encrypted tape contents with a new key, the process is equivalent to restoring the data from tape backup. You decrypt the data with the old DEK and subsequently back up the tape contents to tape storage, which will have the effect of encrypting the data with the new DEK.

Resource allocation

A maximum of ten concurrent rekey sessions are supported per Encryption Group, with a maximum of ten concurrent rekey/encryption sessions per target container and 10 concurrent sessions per physical initiator. If your configuration has two containers that are accessed by the same physical initiator, you cannot have more than ten concurrent rekey or encryption sessions. This includes both rekey (auto and manual) and first-time encryption sessions.

When scheduled rekey or first-time encryption sessions exceed the maximum allowable limit, these sessions will be pending and a **Temporarily out of resources** message is logged. Whenever an active rekey of first-time encryption session completes, the next pending session is scheduled.

The system checks once every 15 minutes to determine if there are any rekey or first-time encryption sessions pending. If resources are available, the next session in the queue is processed. There may be up to an hour lag before the next session in the queue is processed. It is therefore recommended that you do not schedule more than 10 rekey or first-time encryption sessions.

Rekeying modes

Rekeying operations can be performed under the following conditions:

- **Offline rekeying:** The hosts accessing the LUN are offline, or host I/O is halted.
- **Online rekeying:** The hosts accessing the LUN are online, and host I/O is active.

Configuring a LUN for automatic rekeying

Rekeying options are configured at the LUN level either during LUN configuration with the **cryptocfg --add -LUN** command, or at a later time with the **cryptocfg --modify -LUN** command.

For rekeying of a disk array LUN, the Crypto LUN is configured in the following way:

- Set LUN policy as either **cleartext** or **encrypt.**
  - If cleartext is enabled (default), all encryption-related options are disabled and no DEK is associated with the LUN. No encryption is performed on the LUN.
  - If the LUN policy is set to encrypt, encryption is enabled on the LUN and all other options related to encryption are enabled. A DEK is retrieved from the key vault and verified with the metadata.
- Set the auto rekeying feature with the **cryptocfg -enable_rekey** command and specify the interval at which the key expires and automatic rekeying should occur (*time period in days*) Enabling automatic rekeying is valid only if the LUN policy is set to **encrypt** and the encryption format is Brocade **native.** Refer to the section "Crypto LUN parameters and policies" on page 169 for more information.

NOTE
For a scheduled rekeying session to proceed, all encryption engines in a given HA cluster, DEK cluster, or encryption group must be online, and I/O sync links must be configured. Refer to the section "Management LAN configuration" on page 130 for more information.

1. Log in to the group leader as FabricAdmin.
2. Enable automatic rekeying by setting the -**enable_rekey** parameter followed by a time period (in days). The following example enables the automatic rekeying feature on an existing LUN with a 90-day rekeying interval. The data will automatically be re-encrypted every 90 days.

```
FabricAdmin:switch> cryptocfg --modify -LUN my_disk_tgt 0x0 \
10:00:00:00:c9:2b:c9:3a -enable_rekey 90
Operation Succeeded
```

3. Commit the configuration.

```
FabricAdmin:switch> cryptocfg --commit
Operation Succeeded
```

Initiating a manual rekey session

You can initiate a rekeying session manually at your own convenience. All encryption engines in a given HA cluster, DEK cluster, or encryption group must be online for this operation to succeed. The manual rekeying feature is useful when the key is compromised and you want to re-encrypt existing data on the LUN before taking action on the compromised key.

CAUTION

Do not commit this operation if there are any changes pending for the container in which the rekey was started. If you attempt to do this, the system displays a warning stating that the encryption engine is busy and a forced commit is required for the changes to take effect. A forced commit in this situation will halt any rekey that is in-progress (in any container) and corrupt any LUN that is running rekey at the time. There is no recovery for this type of failure.

1. Log in to the group leader as Admin or FabricAdmin.
2. Do LUN discovery by issuing the **cryptocfg --discoverLUN** command (before issuing the **cryptocfg --manual_rekey** command) to avoid a potential I/O timeout because of a path state change at the host.
3. Ensure that all encryption engines in the HA cluster, DEK cluster, or encryption group are online by issuing the **cryptocfg --show -groupmember -all** command.
4. Enter the **cryptocfg --manual_rekey** command. Specify the CryptoTarget container name, the LUN number and the initiator PWWN.

```
FabricAdmin:switch> cryptocfg --manual_rekey my_disk_tgt 0x0\
10:00:00:05:1e:53:37:99
Operation Succeeded
Please check the status of the operation using "cryptocfg --show -rekey"
```

5. Check the status of the rekeying session.

```
FabricAdmin:switch> cryptocfg --show -rekey -all
Number of rekey session(s):     1

Container name:         cx320-157A
EE node:                10:00:00:05:1e:40:4c:00
EE slot:                9
Target:                 50:06:01:60:30:20:db:34 50:06:01:60:b0:20:db:34
Target PID:             022900
VT:                     20:00:00:05:1e:53:8d:cd 20:01:00:05:1e:53:8d:cd
VT PID:                 06c001
Host:                   10:00:00:00:c9:56:e4:7b 20:00:00:00:c9:56:e4:7b
Host PID:               066000
VI:                     20:02:00:05:1e:53:8d:cd 20:03:00:05:1e:53:8d:cd
VI PID:                 06c201
LUN number:             0x1
LUN serial number:
600601603FE2120014FC89130295DF110001000000000000000080000000000000
Rekey session number:   0
Percentage complete:    23
Rekey state:            Write Phase
Rekey role:             Primary/Active
Block size:             512
Number of blocks:       2097152
```

```
Current LBA:            488577
Operation succeeded.
```

Suspension and resumption of rekeying operations

A rekey may be suspended or fail to start for several reasons:

- The LUN goes offline or the encryption switch fails and reboots. Rekey operations are resumed automatically when the target comes back online or the switch comes back up. You cannot abort an in-progress rekey operation.
- An unrecoverable error is encountered on the LUN and the in-progress rekey operation halts. The following LUN errors are considered unrecoverable:

  ```
  SenseKey: 0x3 - Medium Error.
  SenseKey: 0x4 - Hardware Error.
  SenseKey: 0x7 - Data Protect.
  ```

- An unrecoverable error is encountered during the rekey initialization phase. The rekey operation does not begin and a CRITICAL error is logged. All host I/O comes to a halt. All cluster members are notified.
- For any unrecoverable errors that may occur during any other phase of the process, the rekey operation is suspended at that point and a CRITICAL error is logged. All cluster members are notified. Host I/O to all regions of the LUN is halted. Only READ operations are supported for the scratch space region of the LUN used for storing the status block of the rekey operation.

After all errors have been corrected, you have two recovery options:

- **Resume the suspended rekey session.** All DEK cluster or HA cluster members must be online and reachable for this command to succeed. If successful, this command resumes the rekey sessions from the point where it was interrupted.

  1. Log in as Admin or FabricAdmin.
  2. Enter the **cryptocfg** --**resume_rekey** command, followed by the CryptoTarget container name, the LUN number and the initiator PWWN.

     ```
     FabricAdmin:switch> cryptocfg --resume_rekey my_disk_tgt 0x0 \
     10:00:00:05:1e:53:37:99
     Operation Succeeded
     ```

  3. Check the status of the resumed rekey session.

     ```
     FabricAdmin:switch> cryptocfg --show -rekey -all
     ```

- **Read all data off the LUN and write it to another LUN.** In this case, you can cancel the rekey session by removing the LUN from its container and force-committing the transaction. See "Removing a LUN from a CryptoTarget container" on page 173 for instructions on how to remove a LUN by force.

Chapter

4

Deployment Scenarios

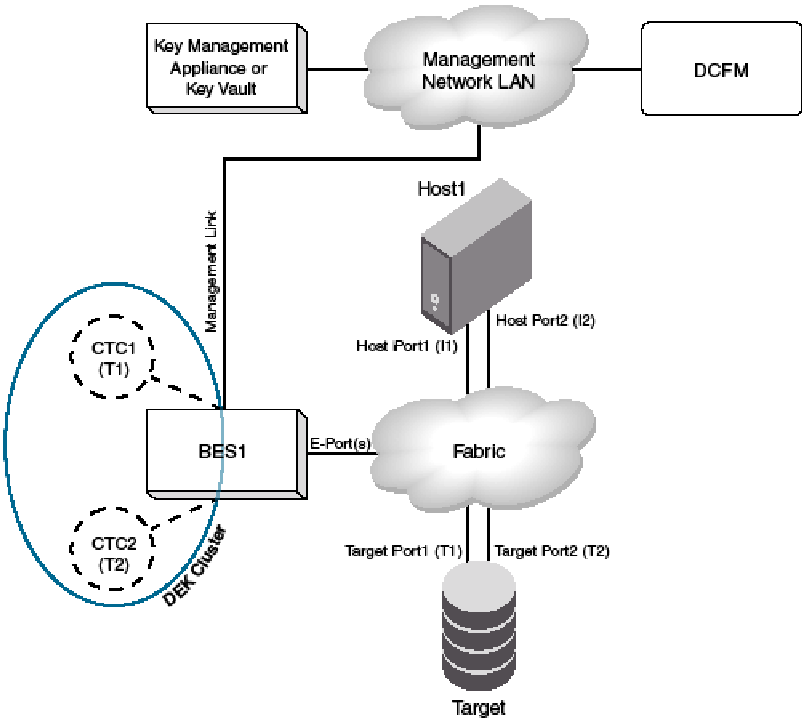
In this chapter

Single encryption switch, two paths from host to target

Figure 94 shows a basic configuration with a single encryption switch providing encryption between one host and one storage device over two the following two paths:

- Host port 1 to target port 1, redirected through CTC T1.
- Host port 2 to target port 2, redirected through CTC T2.

Host port 1 is zoned with target port 1, and host port 2 is zoned with target port 2 to enable the redirection zoning needed to redirect traffic to the correct CTC.



CTC1 - CTC for Target Port T1 hosted on BES1
CTC2 - CTC for Target Port T2 hosted on BES1

FIGURE 94 Single encryption switch, two paths from host to target

Single fabric deployment - HA cluster

Figure 95 shows an encryption deployment in a single fabric with dual core directors and several host and target edge switches in a highly redundant core-edge topology.

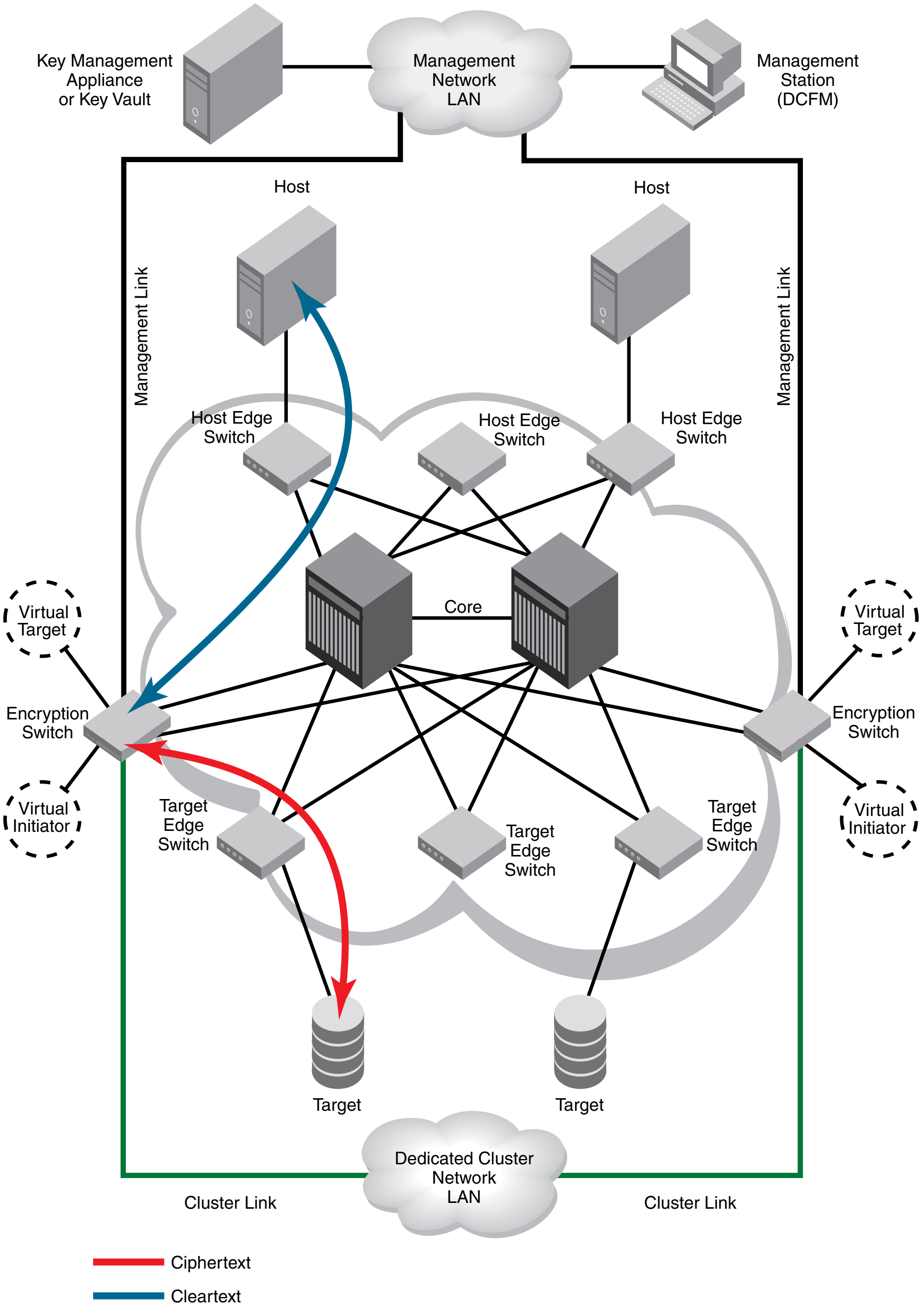


FIGURE 95 Single fabric deployment - HA cluster

In Figure 95, the two encryption switches provide a redundant encryption path to the target devices. The encryption switches are interconnected through a dedicated cluster LAN. The Ge1 and Ge0 gigabit Ethernet ports on each of these switches are attached to this LAN. This LAN connection provides the communication needed to distribute and synchronize configuration information, and enable the two switches to act as a high availability (HA) cluster, providing automatic failover if one of the switches fails, or is taken out of service.

Single fabric deployment - DEK cluster

Figure 96 shows an encryption deployment in a single fabric with two paths between a host and a target.device.

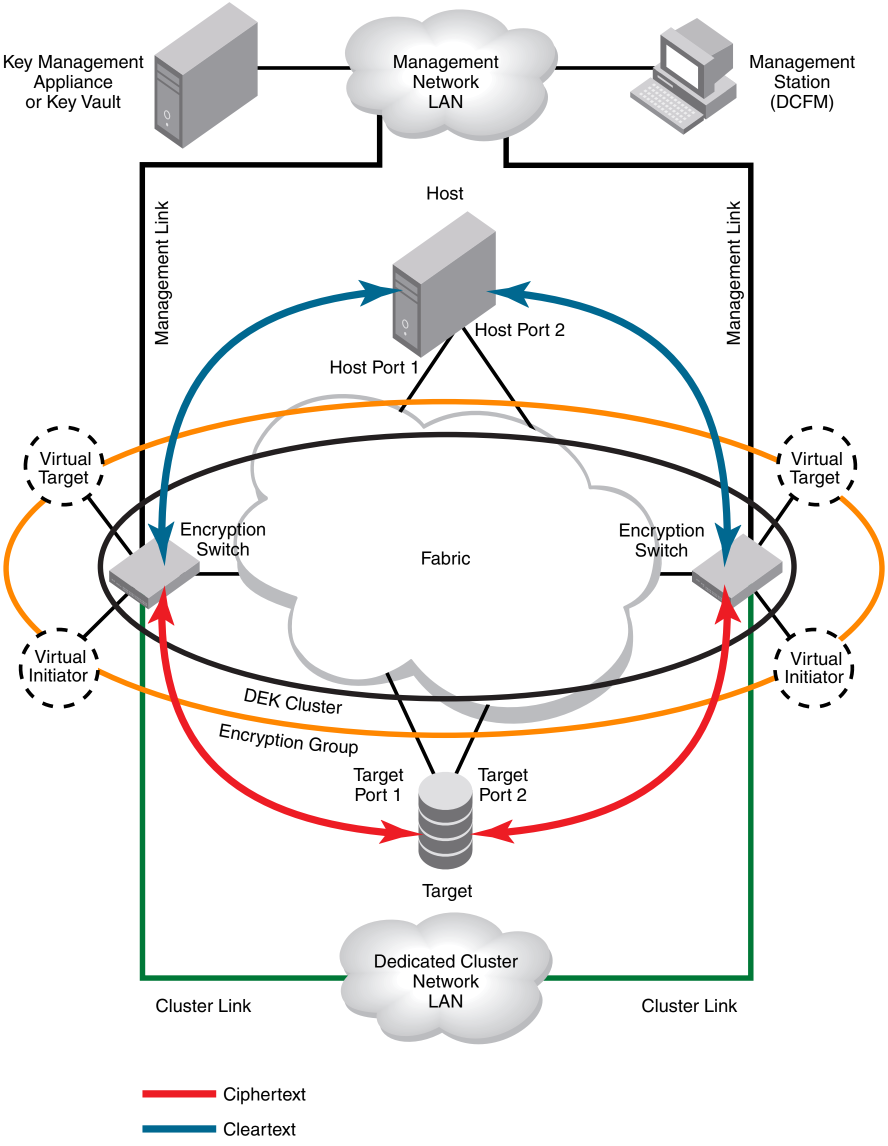


FIGURE 96 Single fabric deployment - DEK cluster

In Figure 96, two encryption switches are required, one for each target path. The path from host port 1 to target port 1 is defined in a CryptoTarget container on one encryption switch, and the path from host port 2 to target port 2 is defined in a CryptoTarget container on the other encryption switch. This forms a DEK cluster between encryption switches for both target paths. The DEK cluster handles the target/host path failover along with the failure of either encryption switch.

Dual fabric deployment - HA and DEK cluster

Figure 97 shows an encryption deployment in a dual fabric SAN. Both fabrics have dual core directors and several host and target edge switches in a highly redundant core-edge topology.

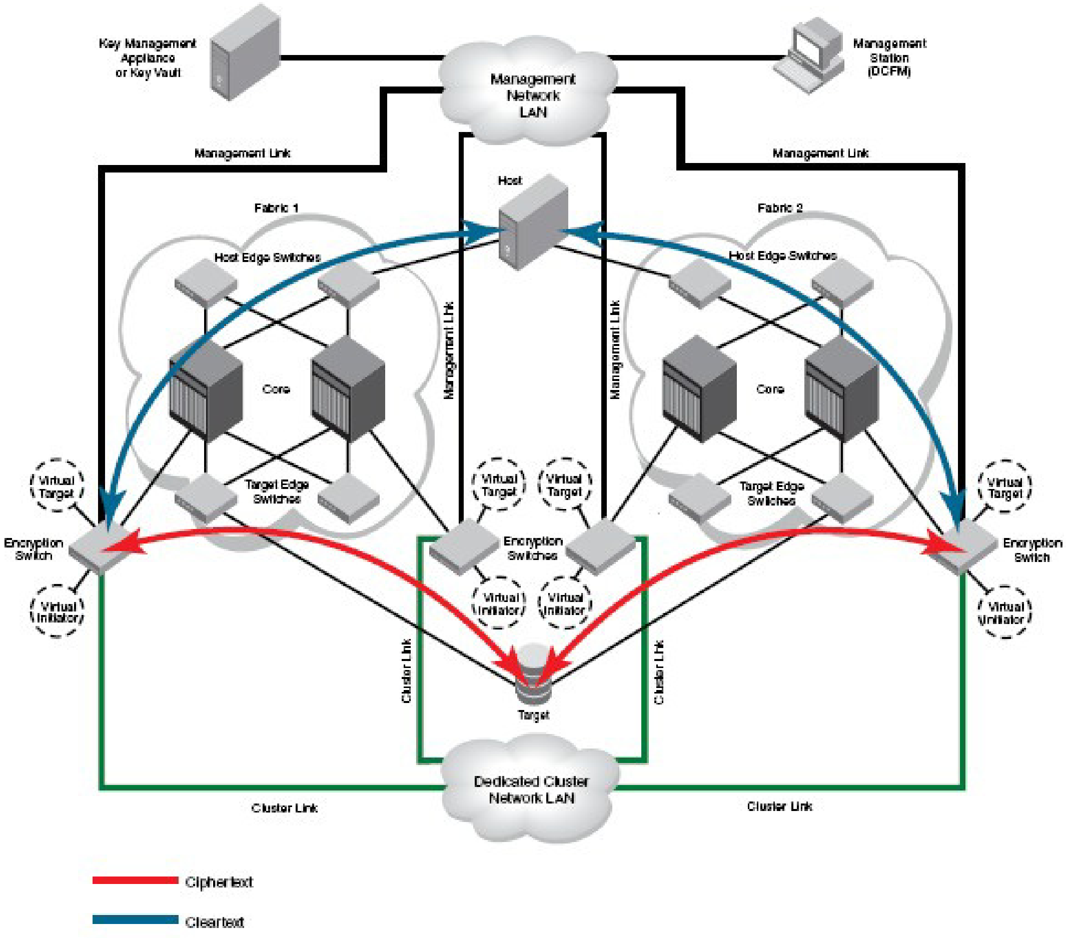


FIGURE 97 Dual fabric deployment - HA and DEK cluster

Figure 97 shows two paths to the target device, one in each fabric. The host also has a path to each fabric. There are two encryption switches in each fabric, interconnected through a dedicated cluster LAN. The Ge1 and Ge0 gigabit Ethernet ports on each of these switches are attached to this LAN. encryption switches 1 and 3 act as a high availability cluster in fabric 1, providing automatic failover for the encryption path between the host and target in fabric 1. Encryption switches 2 and 4 act as a high availability cluster in fabric 2, providing automatic failover for the encryption path between the host and target in fabric 2. All four encryption switches provide an encryption path to the same LUN, and use the same DEK for that LUN, forming a DEK cluster.

Multiple paths, one DEK cluster, and two HA clusters

Figure 98shows a configuration with a DEK cluster that includes two HA clusters, with multiple paths to the same target device.

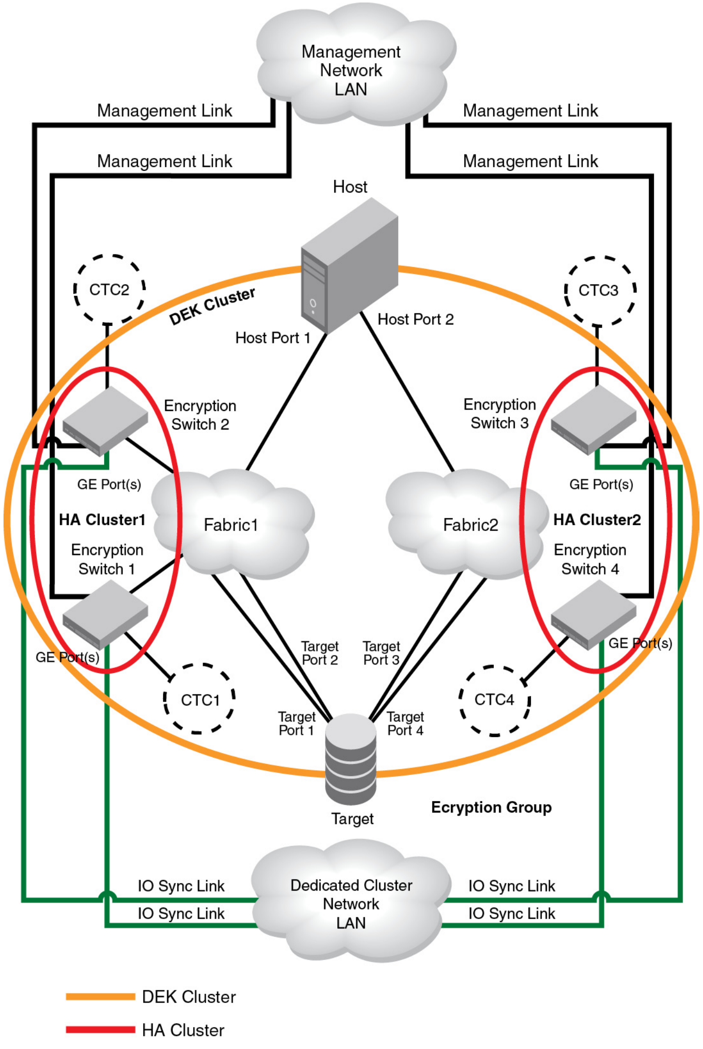


FIGURE 98 Multi-path, DEK cluster and HA cluster

The configuration details shown in Figure 98 are as follows:

- There are two fabrics.
- There are four paths to the target device, two paths in each fabric.
- There are two host ports, one in each fabric.
- Host port 1 is zoned to target port 1 and target port 2 in fabric 1.
- Host port 2 is zoned to target port 3and target port 4 in fabric 2.
- There are four Fabric OS encryption switches organized in HA clusters.
- HA cluster 1 is in fabric 1, and HA cluster 2 is in fabric 2.
- There is one DEK cluster, and one encryption group.

Multiple paths, DEK cluster, no HA cluster

Figure 99 shows a configuration with a DEK cluster with multiple paths to the same target device. There is one encryption switch in each fabric.

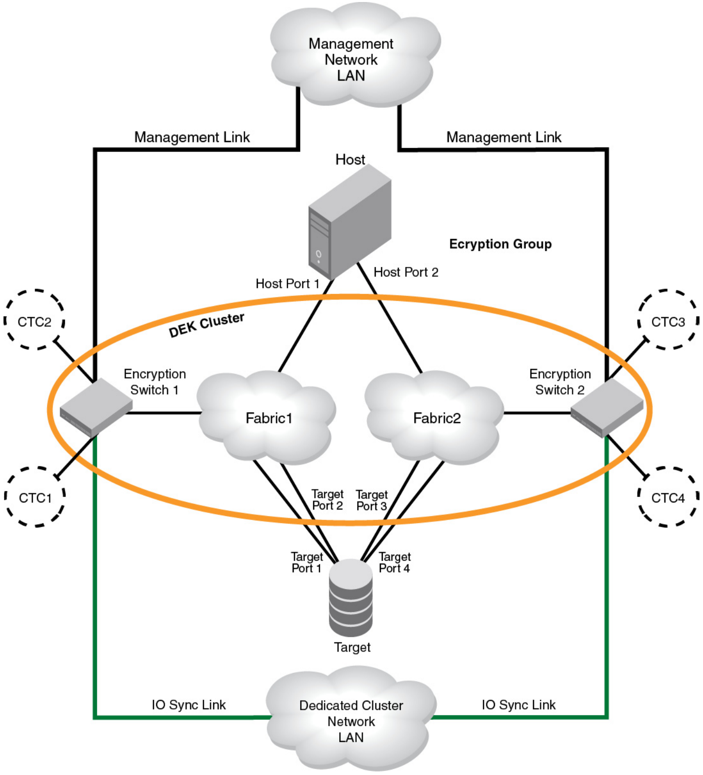


FIGURE 99 **Multi-path, DEK cluster, no HA cluster**

The configuration details are as follows:

- There are two fabrics.
- There are four paths to the target device, two paths in each fabric.
- There are two host ports, one in each fabric.
- Host port1 is zoned to target port1 and target port2 in fabric 1.
- Host port2 is zoned with target port 3 and target port 4 in fabric 2.
- There are two encryption switches, one in each fabric (no HA cluster).
- There is one DEK Cluster and one encryption group.

Deployment in Fibre Channel routed fabrics

In this deployment, the encryption switch may be connected as part of the backbone fabric to another switch or blade that provides the EX_port connections (Figure 100), or it may form the backbone fabric and directly provide the EX_port connections (Figure 101). The encryption resources can be shared with the host and target edge fabrics using device sharing between backbone and edge fabrics.

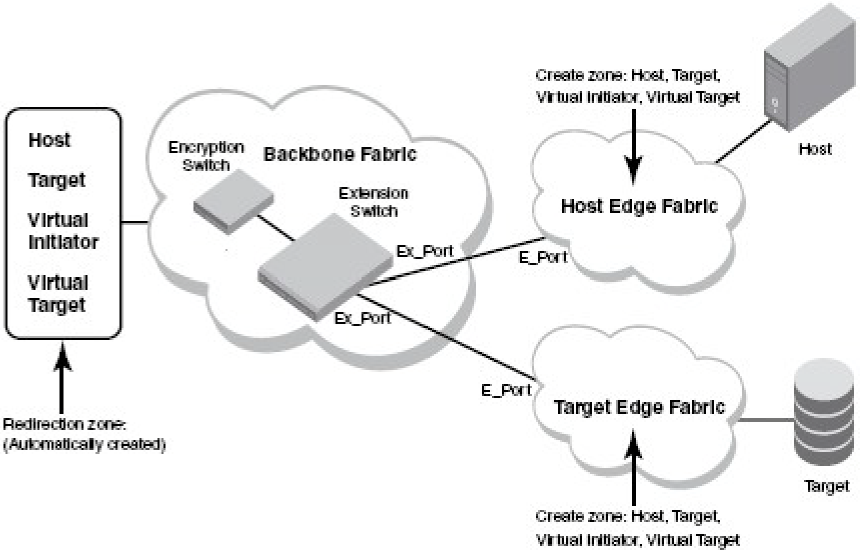

Host
Target
Virtual Initiator
Virtual Target
Redirection zone: (Automatically created)
Encryption Switch
Backbone Fabric
Extension Switch
Ex_Port
Ex_Port
E_Port
E_Port
Create zone: Host, Target, Virtual Initiator, Virtual Target
Host Edge Fabric
Host
Target Edge Fabric
Target
Create zone: Host, Target, Virtual Initiator, Virtual Target

FIGURE 100 Encryption switch connected to FC router as part of backbone fabric

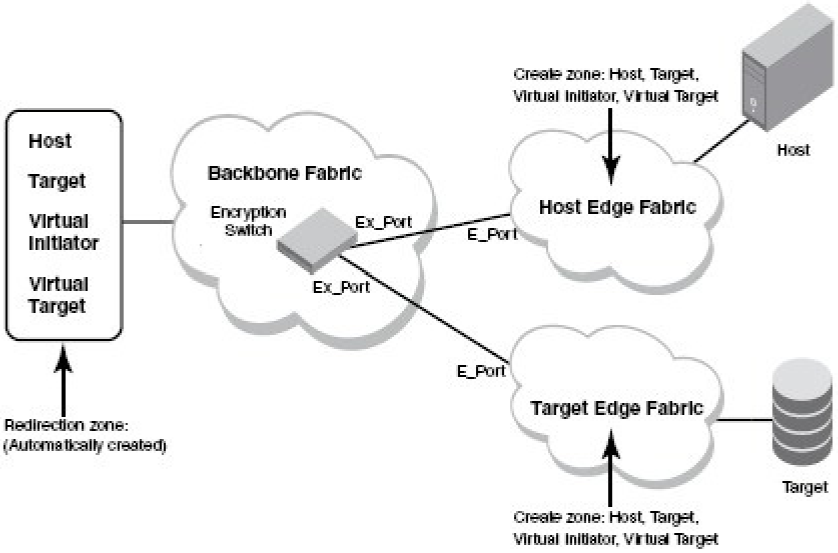

Host
Target
Virtual Initiator
Virtual Target
Redirection zone: (Automatically created)
Backbone Fabric
Encryption Switch
Ex_Port
Ex_Port
E_Port
E_Port
Create zone: Host, Target, Virtual Initiator, Virtual Target
Host Edge Fabric
Host
Target Edge Fabric
Target
Create zone: Host, Target, Virtual Initiator, Virtual Target

FIGURE 101 Encryption switch as FC router and backbone fabric

The following is a summary of steps for creating and enabling the frame redirection zoning features in the FCR configuration (backbone to edge).

- The encryption device creates the frame redirection zone automatically consisting of host, target, virtual target, and virtual initiator in the backbone fabric when the target and host are configured on the encryption device.
- Create the frame redirection zone consisting of host, target, virtual target, and virtual initiator in both the host and target edge fabrics. The CLI command is **zone --rdcreate [host wwn] [target wwn] [VI wwn] [VT wwn][nonrestartable] [FCR]**. Always specify **nonrestartable** as a policy for creating redirection zones. The VI and VT port WWNs can be obtained by running the **cryptocfg --show -container <crypto container name> -cfg** command on the encryption switch or blade. After the redirection zones are created, commit the configuration with the **cfgsave** command.
- Create the LSAN zone consisting of host, target, virtual target, and virtual initiator in both the backbone fabric and the target edge fabrics. Refer to the *Fabric OS Administrator's Guide* for information about LSANs, LSAN zoning, and Fibre Channel routing (FCR) configurations.

Deployment as part of an edge fabric

In this deployment, the encryption switch is connected to either the host or target edge fabric. The backbone fabric may contain a 7800 extension switch or FX8-24 blade in a DCX or DCX 8510 Backbone, or an FCR-capable switch or blade. The encryption resources of the encryption switch can be shared with the other edge fabrics using FCR in the backbone fabric (Figure 102).

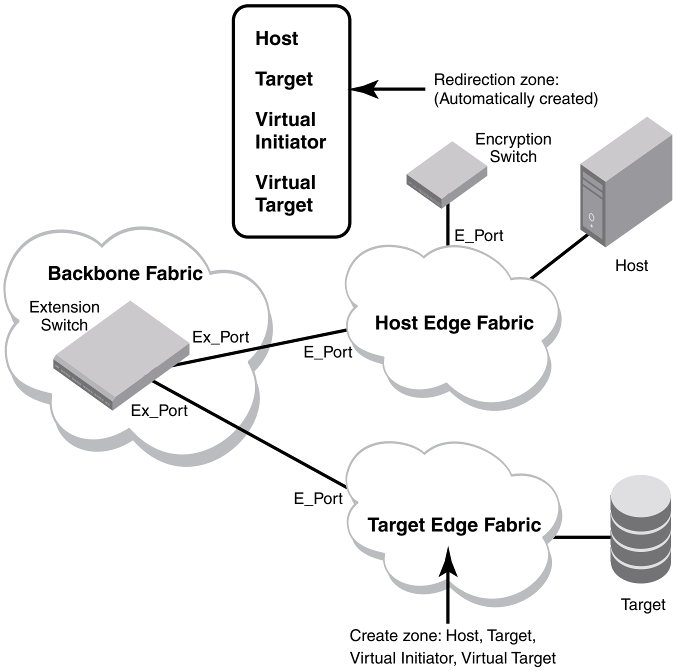


FIGURE 102 **Encryption switch as part of an edge fabric**

The following is a summary of steps for creating and enabling the frame redirection features in the FCR configuration (edge to edge):

- The encryption device creates the frame redirection zone automatically, consisting of host, target, virtual target, and virtual initiator. when the target and host are configured on the encryption device. In Figure 102, the encryption device is connected to the host edge fabric.
- Create the frame redirection one consisting of host, target, virtual target, and virtual initiator in the target edge fabric. The CLI command is **zone --rdcreate [host wwn] [target wwn] [VI wwn] [VT wwn][nonrestartable] [noFCR]**. Always specify **nonrestartable** as policy for creating redirection zones in case of the encryption device. The VI and VT port WWNs can be obtained by running the **cryptocfg --show -container <crypto container name> -cfg** command on the encryption switch or blade. After the redirection zones are created, commit the configuration with the **cfgsave** command.
- Create the LSAN zone consisting of host, target, virtual target, and virtual initiator in both the backbone fabric and the target edge fabrics. Refer to the *Fabric OS Administrator's Guide* for information about LSANs, LSAN zoning, and Fibre Channel routing (FCR) configurations.

Deployment with FCIP extension switches

Encryption switches may be deployed in configurations that use extension switches or extension blades within a DCX or DCX 8510 Backbone to enable long distance connections. Figure 103 shows an encryption switch deployment in a Fibre Channel over IP (FCIP) configuration. Refer to the *Fabric OS Administrator's Guide* for information about creating FCIP configurations.

NOTE
We recommend disabling data compression on FCIP links that might carry encrypted traffic to avoid potential performance issues as compression of encrypted data might not yield the desired compression ratio. We also recommend that tape pipelining and fastwrite also be disabled on the FCIP link if it is transporting encrypted traffic.

When an encryption switch is deployed with an extension switch or blade in the same chassis or fabric, the encryption switch can use the FCIP functionality provided by the extension switch.

In Figure 103, the host is using the remote target for remote data mirroring or backup across the FCIP link. If the encryption services are enabled for the host and the remote target, the encryption switch can take clear text from the host and send cipher text over the FCIP link. For FCIP on the extension switch, this traffic is same as rest of the FCIP traffic between any two FCIP end points. The traffic is encrypted traffic. FCIP provides a data compression option. Data compression should not be enabled on the FCIP link. If compression is enabled on FCIP link, then encrypted traffic going through FCIP compression may not provide the best compression ratio.

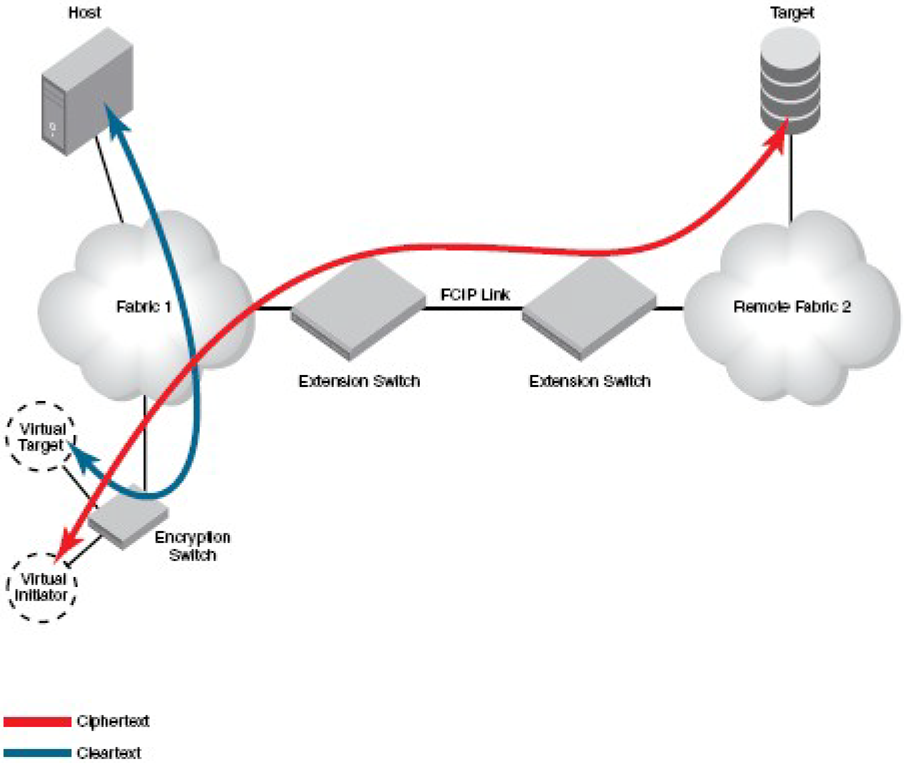


FIGURE 103 FCIP deployment

Data mirroring deployment

Figure 104 shows a data mirroring deployment. In this configuration, the host only knows about target1 and LUN1, and the I/O path to target1 and LUN1. When data is sent to target1, it is written to LUN1, and also sent on to LUN2 for replication. Target1 acts as an initiator to enable the replication I/O path. When an encryption switch is added to the configuration, it introduces another virtual target and LUN, and a virtual initiator in the I/O path in front of target1. The virtual target and LUN provided by the encryption switch is mapped to target1 and LUN1. Data is encrypted and the cipher text is sent to target1, written to LUN1, and replicated on LUN2.

Only one DEK is used to create the cipher text written to both LUNs. A key ID is stored in metadata written to both LUNs. If possible, the metadata is written to every block with the LBA range of 1 to 16. This ensures that the encryption engine will be able to retrieve the correct DEK from the key vault when retrieving data from either LUN1 or LUN2.

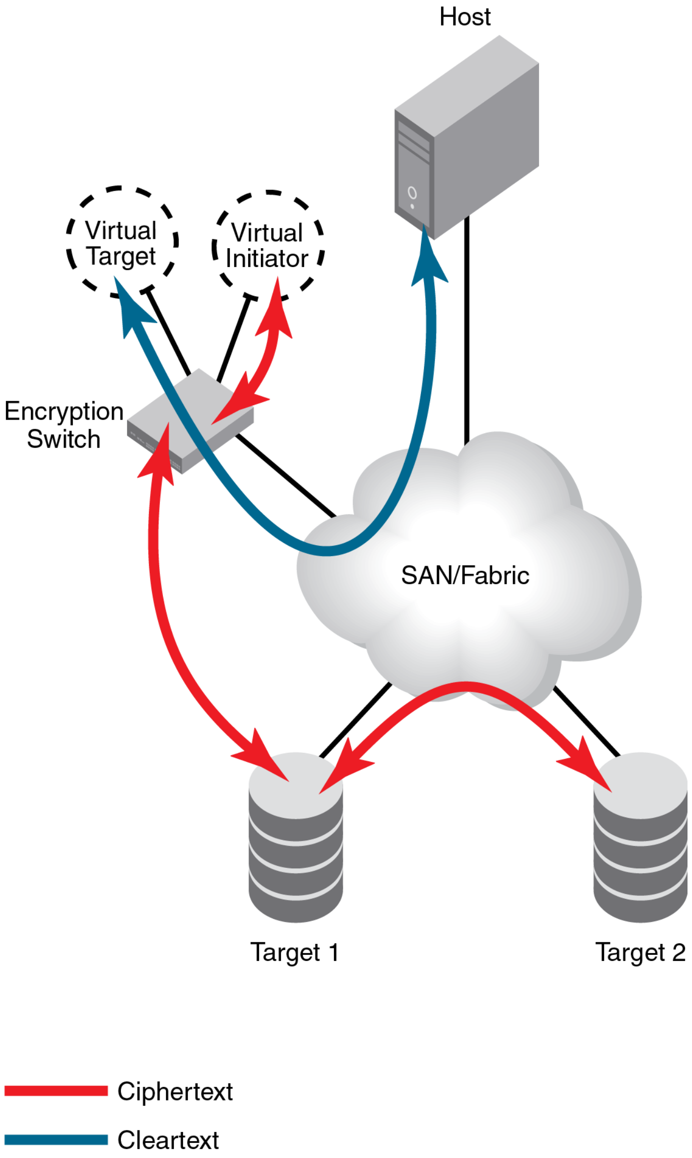


FIGURE 104 Data mirroring deployment

If metadata is not present on the LUN

Beginning with Fabric OS version 6.4.0, this problem is eliminated by enabling the remote replication mode. Remote replication mode may be enabled from either BNA (refer to "Remote replication LUNs" on page 71) or from the command line interface (refer to "Enabling remote replication mode" on page 181).

In very rare cases, when remote replication mode is not enabled, metadata may not be present on the LUN. The record archived in the key vault refers only to the primary LUN, and not to the LUN replication. With no metadata present in the replicated blocks, there is no key ID to use to retrieve the DEK from the key vault. User intervention is needed to query the key vault to get the key ID.

1. Map the primary LUN to the replicated or snapshot LUN.
2. Based on the primary LUN information (mainly target WWN, LUN number, or LUN SN), you can query key records from the key vaults. For this, you need to refer to key management system's documentation to find out how to query key records.
3. Identify the key used during the replication or snapshot of the LUN based on the creation and expiry time of the key at the time the LUN was replicated.
4. When the record is identified, provide the Key ID for the key record as input to the LUN addition for this LUN on the encryption switch or blade. This is done from the key management system's user interface. Refer to the user documentation for the key management system.

VMware ESX server deployments

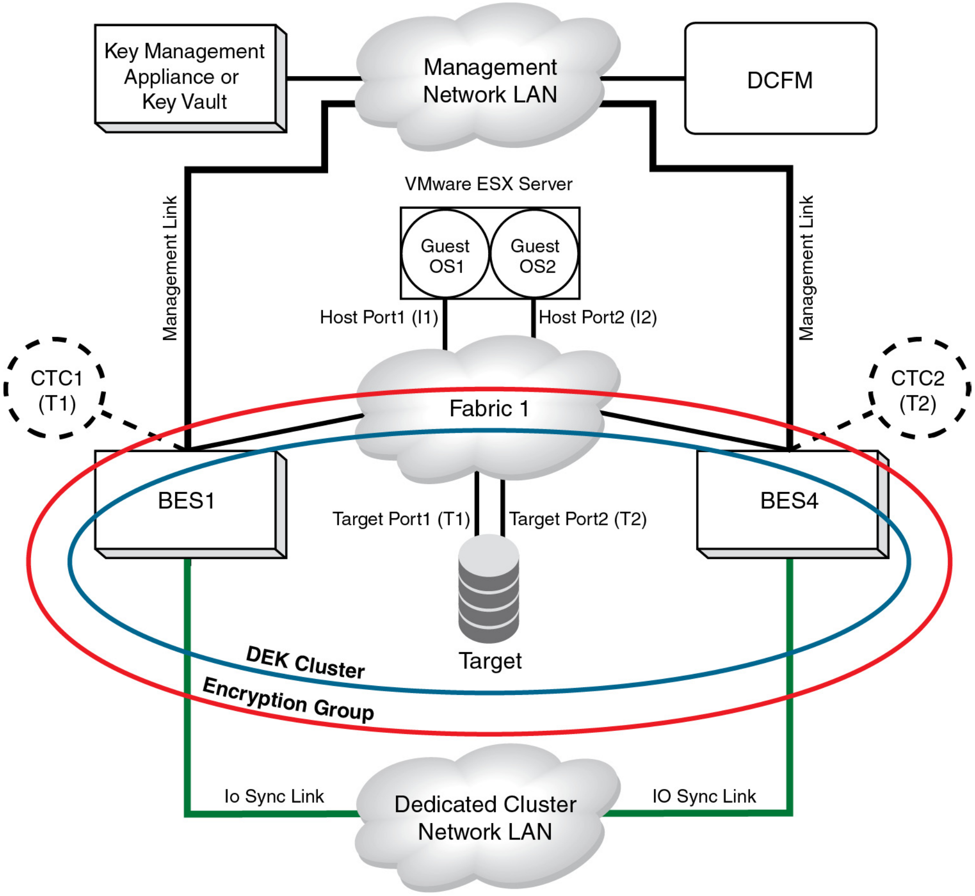
VMware ESX servers may host multiple guest operating systems. A guest operating system may have its own physical HBA port connection, or it may use a virtual port and share a physical HBA port with other guest operating systems.

Figure 105 shows a VMware ESX server with two guest operating systems where each guest accesses a fabric over separate host ports.

There are two paths to a target storage device:

- Host port 1 to target port 1, redirected through CTC T1.
- Host port 2 to target port 2, redirected through CTC T2.

Host port 1 is zoned with target port 1, and host port 2 is zoned with target port 2 to enable the redirection zoning needed to redirect traffic to the correct CTC.



CTC1 - CTC for Target Port T1 hosted on BES1 in DEK Cluster
CTC2 - CTC for Target Port T2 hosted on BES2 in DEK Cluster

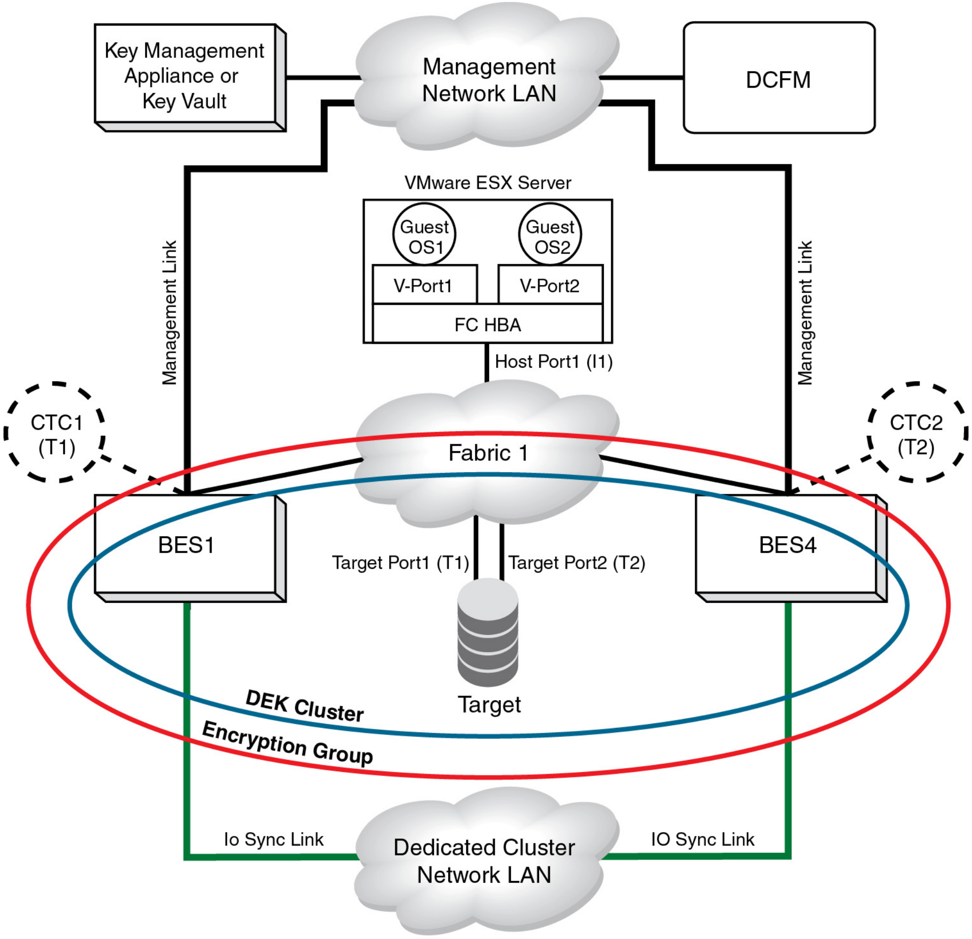
FIGURE 105 VMware ESX server, One HBA per guest OS

Figure 106 shows a VMware ESX server with two guest operating systems where two guests access a fabric over a shared port. To enable this, both guests are assigned a virtual port.

There are two paths to a target storage device:

- Virtual host port 1, through the shared host port, to target port 1, redirected through CTC T1.
- Virtual host port 2, through the shared host port, to target port 2, redirected through CTC T2.

In this case, the virtual host port 1 is zoned with target port 1, and the virtual host port 2 is zoned with target port 2 to enable the redirection zoning needed to redirect traffic to the correct CTC.



CTC1 - CTC for Target Port T1 hosted on BES1 in DEK Cluster
CTC2 - CTC for Target Port T2 hosted on BES2 in DEK Cluster

FIGURE 106 VMware ESX server, One HBA shared by two guest OS

Chapter

Best Practices and Special Topics 5

In this chapter

Firmware upgrade and downgrade considerations

Before upgrading or downgrading firmware, consider the following:

- The encryption engine and the control processor or blade processor are reset after a firmware upgrade. Disruption of encryption I/O can be avoided if an HA cluster is configured. If encryption engines are configured in an HA cluster, perform firmware upgrades one encryption engine at a time so that the partner switch in the HA cluster can take over I/O by failover during a firmware upgrade. When switches form a DEK cluster, firmware upgrades should also be performed one at a time for all switches in the DEK cluster to ensure that a host MPIO failover path is always available.
- If you are upgrading to Fabric OS 7.1.0 from a previous version, (for example, v7.0.1), you must first upgrade the server to DPM 3.2 before you can upgrade to Fabric OS 7.1.0.

NOTE
DPM 3.1 server is not compatible with the earlier client. Refer to the "Fabric OS and DPM Compatibility Matrix" on page 290.

- The following warning can be ignored if the nodes in an EG are running different versions of Fabric OS.
  "2011/04/12-18:41:08, [SPM-1016], 17132, FID 128, WARNING, Security database is out of sync."
- A downgrade to Fabric OS 7.0.1 results in the loss of thin provision LUN information.
- When doing a firmware upgrade to Fabric OS 7.0.0 or downgrade from Fabric OS 7.0.0, the message SPM-1016 will be observed on v7.0.0 nodes in the encryption group (EG) when other nodes in that EG that are still running versions earlier than Fabric OS 7.0.0. Although this is a warning message, it is transient and is only observed during a firmware upgrade or downgrade operation. The message can be ignored.
- Because of the limitations of the RKM 2.1.1 client running Fabric OS 7.0.1 or earlier with the DPM 3.1 server, it is recommended that you upgrade to DPM 3.2 server instead of DPM 3.1.
- Do not try to use the **configUpload** command from Fabric OS v6.4.0 or later and then use the **configDownload** command to Fabric OS v6.3.x or earlier with any Fabric OS v6.4.0 feature in an enabled state.
- You cannot downgrade to a Fabric OS version prior to v6.2.0.

General guidelines

General guidelines for a firmware upgrade of encryption switches and a DCX Backbone chassis with encryption blades in encryption groups, HA clusters, and DEK clusters are as follows:

- Upgrade one node at time.
- Do not perform a firmware upgrade when rekey operations and first-time encryption operations are underway.
- Do not start any manual rekey operations and first-time encryption operations during the firmware upgrade process for all nodes in the HA/DEK cluster.

Guidelines for firmware upgrade of encryption switches and a DCX Backbone chassis with encryption blades deployed in a DEK cluster with two HA clusters:

- Upgrade nodes in one HA cluster at a time.
- Within an HA cluster, upgrade one node at a time.

- Guidelines for firmware upgrade of encryption switches and a DCX Backbone chassis with encryption blades deployed in DEK cluster with No HA cluster (each node hosting one path).
  - Upgrade one node at a time.
  - In the case of active/passive arrays, upgrade the node which is hosting the passive path first. Upgrade the node which is hosting active path next. The Host MPIO ensures that I/O fails over and fails back from active to passive and back to active during this firmware upgrade process.
  - In the case of active/active arrays, upgrade order of nodes does not matter, but you still must upgrade one node at a time. The Host MPIO ensures that I/O fails over and fails back from one active path to another active path during this firmware upgrade process.
- All nodes in an encryption group must be at the same firmware level before starting a rekey or first-time encryption operation.
- A firmware consistency check for Fabric OS 6.4.0(x) and later is enforced in an encryption group if any of the v6.4.0(x) features is enabled, for example, device decommission, disk tape co-existence, and replication. If any Fabric OS 6.4.0(x) feature is in an enabled state, then any firmware download to Fabric OS v6.3.x or earlier is blocked.
  - Do not try registering a node running Fabric OS 6.3.x or earlier to an encryption group when all nodes are running Fabric OS 6.4.0(x) with one or more Fabric OS 6.4.0(x) features enabled.
  - Disable all Fabric OS 6.4.0(x) features before ejecting a node running Fabric OS 6.4.0(x) and registering the node as a member of an encryption group with nodes running Fabric OS 6.3.x or earlier.

Specific guidelines for HA clusters

The following are specific guidelines for a firmware upgrade of the encryption switch or blade when deployed in HA cluster. The guidelines are based on the following scenario:

- There are 2 nodes (BES1 and BES2) in the HA cluster.
- Each node hosts certain number of CryptoTarget containers and associated LUNs.
- Node 1 (BES1) needs to be upgraded first.

1. Change the failback mode to manual if it was set to auto by issuing the following command on the group leader:

```
Admin:switch> cryptocfg --set -failbackmode manual
```

2. On node 1 (BES1), disable the encryption engine to force the failover of CryptoTarget containers and associated LUNs onto the HA cluster peer member node 2 (BES2) by issuing the following command.

```
Admin:switch> cryptocfg --disableEE
```

3. Ensure that these CryptoTarget Containers and LUNs actually fail over to node 2 (BES2) in the HA cluster. Check for all LUNs in encryption enabled state on node 2 (BES2). This ensures that I/O also fails over to node 2 (BES2) and continues during this process.
4. On node 1 (BES1) enable the encryption engine (EE), by issuing the following command.

```
Admin:switch> cryptocfg --enableEE
```

5. Start firmware download (upgrade) on the node 1 (BES1). Refer to the *Fabric OS Administrator's Guide* to review firmware download procedures.
6. After firmware download is complete and node 1 (BES1) is back up, make sure the encryption engine is online.
7. On node 1 (BES1) initiate manual failback of CryptoTarget containers and associated LUNs from node 2 (BES2) to node 1 (BES1) by issuing the following command.

   ```
   Admin:switch> cryptocfg --failback -EE
   ```

8. Check that CryptoTarget Containers and associated LUNs fail back successfully on node 1 (BES1), and host I/O also moves from node 2 (BES2) to node 1 (BES1) and continues during the failback process.
9. To upgrade node 2 (BES2), Repeat steps 2 to 8.
10. After all nodes in the Encryption Group have been upgraded, change back the failback mode to auto from manual, if required, by issuing the following command.

    ```
    Admin:switch> cryptocfg --set -failback auto
    ```

Configuration upload and download considerations

Security information is not included when you upload a configuration from an encryption switch or blade. Extra steps are necessary before and after download to re-establish that information. The following sections describe what information is included in a upload from an encryption group leader and encryption group member load, what information is not included, and the steps to take to re-establish the information.

Configuration upload at an encryption group leader node

A configuration upload performed at an encryption group leader node contains the following:

- The local switch configuration.
- Encryption group-related configuration.
- The encryption group-wide configuration of CryptoTargets, disk and tape LUNs, tape pools, HA clusters, security, and key vaults.

Configuration upload at an encryption group member node

A configuration upload at an individual encryption group member node contains the following:

- The local switch configuration.
- Encryption group-related configuration.
- Encryption group-wide configuration of CryptoTargets, disk and tape LUNs, tape pools, HA clusters, security, and key vaults.

Information not included in an upload

The following certificates will be not be present when the configuration is downloaded:

- External certificates imported on the switch:
  - key vault certificate
  - peer node/switch certificate
  - authentication card certificate
- Certificates generated internally:
  - KAC certificate
  - CP certificate
  - FIPS officer and user certificates

The Authentication Quorum size is included in the configuration upload for read-only purposes, but is not set by a configuration download.

Steps before configuration download

The configuration download does not have any certificates, public or private keys, master key, or link keys included. Perform following steps prior to configuration download to generate and obtain the necessary certificates and keys:

1. Use the following commands to initialize the encryption engine

   ```
   cryptocfg --InitNode
   cryptocfg --initEE
   cryptocfg --regEE
   ```

   Initializing the switch generates the following internal certificates:

   - KAC certificate
   - CP certificate
   - FIPS officer and user certificates

2. Import peer nodes/switches certificates onto the switch prior to configuration download.
3. Import key vault certificates onto switch prior to configuration download.
4. Create an encryption group with same name as in configuration upload information for the encryption group leader node.
5. Import Authentication Card Certificates onto the switch prior to configuration download.

Configuration download at the encryption group leader

The configuration download contains the encryption group-wide configuration information about CryptoTargets, disk and tape LUNs, tape pools, HA clusters, security, and key vaults. The encryption group leader first applies the encryption group-wide configuration information to the local configuration database and then distributes the configuration to all members in the encryption group. Also any layer-2 and switch specific configuration information is applied locally to the encryption group leader.

Configuration download at an encryption group member

Switch specific configuration information pertaining to the member switch or blade is applied. Information specific to the encryption group leader is filtered out.

Steps after configuration download

For all opaque key vaults, restore or generate and backup the master key. In a multiple node encryption group, the master key is propagated from the group leader node.

1. Use the following command to enable the encryption engine.

   ```
   Admin:switch> cryptocfg --enableEE [slot num]
   ```

2. Commit the configuration.

   ```
   Admin:switch> cryptocfg --commit
   ```

3. If there are containers that belonged to the old encryption switch or blade, then after **configdownload** is run, use the following command to change the ownership of containers to the new encryption switch or blade, assuming the host and target physical zone exists.

   ```
   Admin:switch> cryptocfg --replace  <old EE WWN> <new EE WWN>
   ```

4. Commit the configuration.

   ```
   Admin:switch> cryptocfg --commit
   ```

5. Use the following command to check if the switch or blade has the master key.

   ```
   Admin:switch> cryptocfg --show -groupmember <switch WWN>
   ```

6. If a master key is not present, restore the master key from backed up copy. Procedures will differ depending on the backup media used (from recovery smart cards, from the key vault, from a file on the network or a file on a USB-attached device). If new master key needs to be generated, generate the master key and back it up.

If authentication cards are used, set the authentication quorum size from the encryption group leader node after importing and registering the necessary number of Authentication Card certificates.

HP-UX considerations

The HP-UX OS requires LUN 0 to be present. LUNs are scanned differently based on the type value returned for LUN 0 by the target device.

- If the type is 0, then HP-UX only scans LUNs from 0 to 7. That is the maximum limit allowed by HP-UX for device type for type 0.
- If the type is 0xC, then HP-UX scans all LUNs.

For HP-UX multi-path configurations:

- Add LUN 0 as a cleartext LUN.
- Make sure to configure a dummy LUN 0 for each host accessing multi-path LUNs through CTCs in the encryption switch.

```
cryptocfg --add –LUN <crypto target container name> 0 <initiator PWWN>
<initiator NWWN>
```

Best practices are as follows:

- Create a cryptoTarget container for the target WWN.
- Add the HP-UX initiator WWN to the container.
- Issue the discover LUN CLI command on the container to discover the LUNs present in the target.
- Based on the LUN list returned as part of LUN discovery, add the LUN 0 if LUN 0 is present in the target (which is usually the case).

NOTE
When an EMC-CX3 storage array is used with HP-UX the CX3 array exposes both 0x0 and 0x4000 LUNs to the HP-UX host. 0x0 and 0x4000 LUNs have the same LSN. Both must be added as cleartext.

AIX Considerations

For AIX-based PowerHA SystemMirror host clusters, the cluster repository disk should be defined outside of the encryption environment.

Ensure that Dynamic Tracking is set to “Yes” for all Fibre Channel adapters on the AIX system.

Enabling a disabled LUN

When Metadata is found on the LUN, but current LUN state is indicated as cleartext or is being converted from encrypt to cleartext, the LUN is disabled and the LUN status displayed by the LUN Show CLI command is Internal EE LUN state: Encryption disabled <Reason Code>.

The disabled LUN can be enabled by invoking the enable LUN command.

```
switch:admin> cryptocfg --enable -LUN <crypto target container name> <LUN Num>
<InitiatorPWWN>
```

Decommissioning in an EG containing mixed modes

If you have an encryption group (EG) that contains mixed nodes, (for example, one member node is running Fabric OS 7.0.0 and another member node is running Fabric OS 6.4.2), you might notice that after you decommission a LUN, the decommissioned Key IDs might not be displayed on the node running v6.4.2, even though the decommission operation was successful.

In a mixed encryption group consisting of nodes running Fabric OS 7.0.0 and an earlier Fabric OS version, such as 6.4.x, the decommission operation will complete successfully and the LUNs will be removed from the hosted containers; however, the list of decommissioned key IDs might not be displayed correctly from all nodes in the encryption group. To resolve this, ensure that the Fabric OS version running on all nodes in an encryption group is the same version. Otherwise some of the crypto commands might not work as expected.

Decommissioning a multi-path LUN

When issuing a decommission command on a multi-path LUN on the Group Leader of an encryption group, make sure you do not issue a second decommission request from another path to the same LUN from a member node. Doing so causes a timeout with an accompanying message, "EE(s) is busy. Please try it later."

You can avoid this scenario by making sure that a second decommission operation is not requested on a node where the LUN state is shown as “Commit in progress.”

If you are in a position whereby you receive the error message, simply re-issue the decommission request.

Disk metadata

If possible, 32 bytes of metadata are added to every block in LBA range 1 to 16 for both the native Brocade format and DF-compatible formats. This metadata is not visible to the host. The Host I/Os for the metadata region of the LUN are handled in the encryption switch software, and some additional latency should be expected.

NOTE
For encrypted LUNs, data in LBA 0 will always be in cleartext.

Tape metadata

One kilobyte of metadata is added per tape block for both the native Brocade format and DF-compatible formats. Tape block size (as configured by host) is modified by the encryption device to accommodate 1K metadata per block. A given tape can have a mix of compressed and uncompressed blocks. Block lengths are as follows.

Encrypted/Compressed Tape Block Format	Compressed and encrypted tape block data + 1K metadata + ASCII 0 pad = block length of tape.
Encrypted Tape Block Format (No Compression)	Encrypted tape block data + 1K metadata = block length of tape.

Tape data compression

Data is compressed by the encryption switch or blade before encrypting only if the tape device supports compression, and compression is explicitly enabled by the host backup application. That means if the tape device supports compression, but is not enabled by the host backup application, then compression is not performed by the encryption switch or blade before encrypting the data. However, if the backup application turns on compression at the tape device and does not turn it off before logout or after the backup or restore operation is complete, and a second host backup application starts using the same tape device and does not explicitly turn off compression, compression will still be on when the encryption switch or blade issues a Mode Sense command to find target device capabilities, and compression is used. In other words, if the host backup application does not turn off compression on the target, the encryption switch or blade uses the compression feature of the target. Conversely, if the tape device does not support compression, the encryption switch or blade does not perform compression before encrypting the data. The same rules apply for decompression.

Data is compressed, encrypted and padded with ASCII 0 to the tape block length to simplify handling at the encryption device. It is assumed that a tape target with compression enabled will be unable to compress the seemingly random encrypted data, but will greatly compress the padded zero data that follows. Compressing data at the encryption device in conditions other than above does not create any additional space savings on the tape media.

Tape pools

When a new tape pool needs to be created, the following steps must be performed:

- Configure the tape pool with a maximum of 64 bytes of tape pool label first on the encryption device. The tape pool label configured on the encryption device must be an exact match to the tape pool label (or number) configured on the tape backup application.
- Set the policies (such as encrypt or cleartext), format (such as native Brocade format or DF-compatible), and optionally specify a key life span for the tape pool.

Tape pools are unique across an encryption group. Tape pool configuration takes precedence over LUN level configuration.

Tape pool configuration is used only when labeling of tape media is done on the first write for the tape media. After tape labeling is done and metadata written, the tape pool configuration is no longer used. Tape pool configuration is not required for restoring data from the encrypted tape belonging to the tape pool, because the key ID is present in the metadata.

When the tape pool label configured on the encryption device does not match with any label that the backup application sends as part of the first write (tape labeling) to the tape media, the tape pool level policies are ignored and default LUN level policies are applied.

Tape block zero handling

The block zero of the tape media is not encrypted and the data in the block zero is sent as cleartext along with the block zero metadata header prefixed to the data to the tape device.

Tape key expiry

When the tape key of native pools expires in the middle of a write operation on the tape, the key is used for the duration of any write operation to append the data on the tape media. On any given tape medium, the same key is used for all written blocks, regardless of the time in between append operations.

With the exception of native pools, whenever you rewind a tape and write to block zero, a new key will be generated that is unique to that tape. Only with native pools will the same key be used to write to multiple media. This key has a user-determined lifespan, which applies to the elapsed time between write operations to new tapes (after rewind).

Note the following:

- Key expiration does not apply to append operations, no matter how long in the future.
- Key expiration never applies to read operations.
- Key expiration never applies to LUN-based policies. A new key is generated every time a tape media is rewound and written to block zero (label), regardless of whether the specified key life span has expired.

Configuring CryptoTarget containers and LUNs

The following are best practices to follow when configuring CryptoTarget containers and crypto LUNs:

- Host a target port on only one encryption switch, or one HA cluster. All LUNs visible through the target port are hosted on the same encryption switch, and are available for storing cipher text.
- Be sure all nodes in a given DEK or HA cluster are up and enabled before creating an encrypted LUN. If a node in the DEK or HA cluster is down, or the encryption engine is down or not enabled when an encrypted LUN is added to the CryptoTarget container, write operations will hang when writing metadata to the LUN, and I/O will timeout. Data integrity is not guaranteed in this condition.
- Before committing CryptoTarget container or LUN configurations or modifications on an encryption switch or FS8-18 blade, make sure that there are no outstanding zoning transactions in the switch or fabric. If there is an outstanding zoning transaction, the commit operation will fail and result in disabling the LUN. You can check for outstanding zoning transactions by issuing the **cfgtransShow** command.
- LUNs are uniquely identified by the encryption switch or FS8-18 blade using the LUN serial number. The LUN serial number must be unique for LUNs exposed from the same target port. The LUN serial number must be unique for LUNs belonging to different target ports in non-multipathing configurations. Failure to ensure that the serial numbers are unique will result in undefined behavior and may result in faulting the encryption switch or FS8-18 blade.

- To enable host MPIO, LUNs must also be available through a second target port, hosted on a second encryption switch, the same encryption switch or encryption engine. The second encryption switch could be in the same fabric, or a different fabric.
- Hosts should be able to access LUNs through multiple ports for redundancy.
- For high availability and failover within the fabric, implement an HA cluster of two encryption switches, and host the target port as a virtual target on one of the switches.
- Don't change the WWN of any node after it has been deployed in an encryption group.
- To minimize host IO disruption or time-outs during CryptoTarget container failover, it is recommended that the devices (hosts, target ports) are connected to an edge switch in a fabric, and not directly to Encryption switch/blade ports.
- Always use the following process when configuring the LUN for encryption, unless the LUN was previously encrypted.

  1. Add the LUN as **cleartext** to the CryptoTarget container.
  2. When the LUN comes online and Host I/O starts flowing through the LUN as cleartext, then modify the LUN from cleartext to **encrypt** and **enable_encexistingdata** options to convert the LUN to encryption.

  An exception to this LUN configuration process is that if the LUN was previously encrypted by the encryption switch or FS8-18 blade, then the LUN can be added to the CryptoTarget Container with the –**encrypt** and –**lunstate encrypted** options.

Redirection zones

Redirection zones should not be deleted. If a redirection zone is accidentally deleted, I/O traffic cannot be redirected to encryption devices, and encryption is disrupted. To recover, re-enable the existing device configuration by invoking the **cryptocfg** --**commit** command on the group leader. If no changes have taken place since the last commit, you should use the **cryptocfg** --**commit** –**force** command. This recreates redirection zones related to the device configuration in the zone database, and restores frame redirection, which makes it possible to restore encryption.

To remove access between a given initiator and target, remove both the active zoning information between the initiator and target, and the associated CryptoTarget Containers (CTCs). This will remove the associated frame redirection zone information.

Deployment with Admin Domains (AD)

Virtual devices created by the encryption device do not support the AD feature in this release. All virtual devices are part of AD0 and AD255. Targets for which virtual targets are created and hosts for which virtual initiators are created must also be in AD0 and AD255. If they are not, access from the hosts and targets to the virtual targets and virtual initiators is denied, leading to denial of encryption services.

Do not use DHCP for IP interfaces

Do not use DHCP for either the GbE management interface or the Ge0 and Ge1 interfaces. Assign static IP addresses.

Ensure uniform licensing in HA clusters

Licenses installed on the nodes should allow for identical performance numbers between HA cluster members.

Tape library media changer considerations

In tape libraries where the media changer unit is addressed by a target port that is separate from the actual tape SCSI I/O ports, create a CryptoTarget container for the media changer unit and CryptoTarget containers for the SCSI I/O ports. If a CryptoTarget container is created only for the media changer unit target port, no encryption is performed on this device.

In tape libraries where the media changer unit is addressed by separate LUN at the same target port as the actual tape SCSI I/O LUN, create a CryptoTarget container for the target port, and add both the media changer unit LUN and one or more tape SCSI I/O LUNs to that CryptoTarget container. If only a media changer unit LUN is added to the CryptoTarget container, no encryption is performed on this device.

Turn off host-based encryption

If a host has an encryption capability of any kind, be sure it is turned it off before using the encryption engine on the encryption switch or blade. Encryption and decryption at the host may make it impossible to successfully decrypt the data.

Avoid double encryption

Encryption and decryption at tape drives does not affect the encryption switch or blade capabilities, and does not cause problems with decrypting the data. However, double encryption adds the unnecessary need to manage two sets of encryption keys, increases the risk of losing data, may reduce performance, and does not add security.

PID failover

Virtual device PIDs do not persist upon failover within a single fabric HA cluster. Upon failover, the virtual device is s assigned a different PID on the standby encryption switch or blade.

Some operating systems view the PID change as an indication of path failure, and will switch over to redundant path in another fabric. In these cases, HA clusters should not be implemented. These operating systems include the following:

- HP-UX prior to 11.x. The issue is not present beginning with 11.31 and later releases.
- All versions of IBM AIX, unless dynamic tracking is enabled.
- Solaris 2.x releases, Solaris 7, and later releases.

Turn off compression on extension switches

We recommend disabling data compression on FCIP links that might carry encrypted traffic to avoid potential performance issues as compression of encrypted data might not yield desired compression ratio. We also recommend that tape pipelining and fastwrite also be disabled on the FCIP link if it is transporting encrypted traffic.

Rekeying best practices and policies

Rekeying should be done only when necessary. In key management systems, DEKs are never exposed in an unwrapped or unencrypted state. For all opaque key management systems, you must rekey if the master key is compromised. The practice of rekeying should be limited to the following cases:

- Master key compromise in the case of opaque key vaults.
- Insider security breaches.
- As a general security policy as infrequently as every six months or once per year.

Manual rekey

Ensure that the link to the key management system is up and running before you attempt a manual rekey.

Latency in rekey operations

Host I/O for regions other than the current rekey region has no latency during a rekey operation. Host I/O for the region where the current rekey is happening has minimal latency (a few milliseconds) because I/O is held until the rekey is complete. The I/O sync links (the Ethernet ports labeled Ge0 and Ge1) must be configured, and must both be connected to the I/O sync LAN to enable proper handling of rekey state synchronization in high availability (HA cluster) configurations.

Allow rekey to complete before deleting a container

Do not delete a crypto container while rekey is in session or if rekey is not completed. If you want to delete a container, use the command **cryptocfg --show -rekey -all** to display the status of rekey sessions. If any rekey session is not 100% completed, do not delete the container. If you do delete the container before rekey is complete, and subsequently add the LUN back as cleartext, all data on the LUN is destroyed.

Rekey operations and firmware upgrades

All nodes in an encryption group must be at the same firmware level before starting a rekey or first-time encryption operation. Make sure that existing rekey or first-time encryption operations complete before upgrading any of the encryption products in the encryption group, and that the upgrade completes before starting a rekey or first-time encryption operation.

Do not change LUN configuration while rekeying

Never change the configuration of any LUN that belongs to a CryptoTarget container/LUN configuration while the rekeying process for that LUN is active. If you change the LUN's settings during manual or auto, rekeying or first-time encryption, the system reports a warning message stating that the encryption engine is busy and a forced commit is required for the changes to take effect. A forced commit command halts all active rekeying progresses running in all CryptoTarget containers and corrupts any LUN engaged in a rekeying operation. There is no recovery for this type of failure.

Recommendation for Host I/O traffic during online rekeying and first-time encryption

You may see failed I/Os if writes are done to a LUN that is undergoing first-time encryption or rekeying. It is recommended that host I/O operations are quiesced and not started again until rekey operations or first-time encryption operations for the LUN are complete.

KAC certificate registration expiry

It is important to keep track as to when your signed KAC certificates will expire. Failure to work with valid certificates causes certain commands to not work as expected. If you are using the certificate expiry feature and the certificate expires, the key vault server will not respond as expected. For example, the Group Leader in an encryption group might show that the key vault is connected; however, a member node reports that the key vault is not responding.

To verify the certificate expiration date, use the following command:

```
openssl x509 –in signed_kac_cert.pem -dates –noout

Output:
            Not Before: Dec  4 18:03:14 2009 GMT
            Not After : Dec  4 18:03:14 2010 GMT
```

In the example above, the certificate validity is active until "Dec 4 18:03:14 2010 GMT." After the KAC certificate has expired, the registration process must be redone.

NOTE
In the event that the signed KAC certificate must be re-registered, you will need to log in to the key vault web interface and upload the new signed KAC certificate for the corresponding Brocade Encryption Switch identity.

You can change the value of the certificate expiration date using the following command:

```
openssl x509 -req -sha1 -CAcreateserial -in certs/<Switch CSR Name> -days 365 -CA
cacert.pem -CAkey private/cakey.pem -out newcerts/<Switch Cert Name>
```

In the example above, the certificate is valid for a period of one year (365 days). You can increase or decrease this value according to your own specific needs. The default is 3649 days, or 10 years.

Changing IP addresses in encryption groups

Generally, when IP addresses are assigned to the Ge0 and Ge1 ports, they should not be changed. If an encryption group member node IP address must be changed, refer to "IP Address change of a node within an encryption group" on page 132.

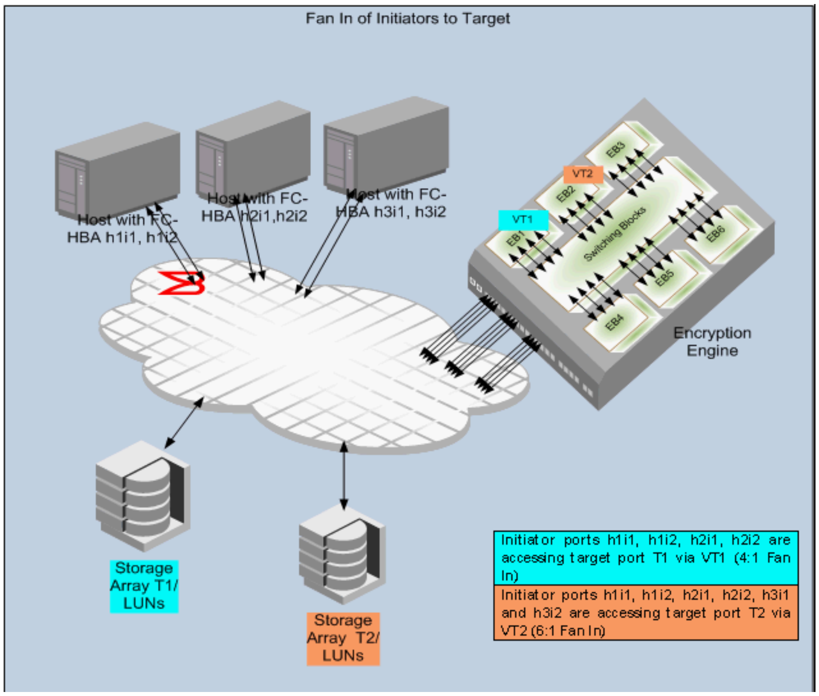
Disabling the encryption engine

The disable encryption engine interface command **cryptocfg --disableEE [slot number]** should be used only during firmware download, and when the encryption and security capabilities of the encryption engine have been compromised. When disabling the encryption capabilities of the encryption engine, be sure the encryption engine is not hosting any CryptoTarget containers. All CryptoTarget containers hosted on the encryption switch or FS8-18 blade must either be removed from the encryption engine, or be moved to different encryption engine in an HA Cluster or encryption group before disabling the encryption and security capabilities.

Recommendations for Initiator Fan-Ins

For optimal performance at reasonable scaling factors of initiators, targets, and LUNs accessed, Brocade Encryption Engines (EEs) are designed to support a fan-in ratio of between four and eight initiator ports to one target port, in terms of the number of distinct initiator ports to a Crypto Container (i.e., a virtual target port corresponding to the physical target port).

An encryption engine has 6 distinct encryption blocks with 4 ports, each port operating at 4 Gbps. The architecture of the encryption blocks provides the potential for an aggregate 96 Gbps of full duplex encryption bandwidth, if the performance license is installed. Figure 107 shows the encryption blocks within an encryption engine, and the host initiator to target port fan-ins. Each encryption engine can host up to 256 distinct targets with a mapping of 1024 initiators accessing all the targets. This brings the fan-in ratio for each target to be 1:4 initiators.


Fan In of Initiators to Target
Host with FC-HBA h1i1, h1i2
Host with FC-HBA h2i1,h2i2
Host with FC-HBA h3i1, h3i2
VT1
VT2
EB1
EB2
EB3
EB4
EB5
EB6
Switching Blocks
Encryption Engine
Storage Array T1/ LUNs
Storage Array T2/ LUNs

Initiator ports h1i1, h1i2, h2i1, h2i2 are accessing target port T1 via VT1 (4:1 Fan In)

Initiator ports h1i1, h1i2, h2i1, h2i2, h3i1 and h3i2 are accessing target port T2 via VT2 (6:1 Fan In)

FIGURE 107 Fan-in ratios with performance license installed

The fan-in ratio for a target can be higher depending on the maximum bandwidth accepted by the target. If the I/O throughput across all initiator ports accessing the target port is well balanced, it is recommended that the maximum fan-in ratio be kept to 8 Initiator ports to 1 target port for optimal performance. Note that this recommendation holds for initiators running at 4 Gbps or less. If a mix of 8 Gbps and other 4 Gbps or less initiator is used, then the maximum fan-in will depend on the maximum sustained bandwidth these initiators would be pushing together over the link to the same target port and across all the target ports hosted on a given encryption engine.

NOTE

If the performance license is not installed, 48 Gbps of full duplex encryption bandwidth is available on the encryption engine, Each of the six encryption blocks will use two ports instead of four, reducing the fan-in ratio by a factor of two.

Best practices for host clusters in an encryption environment

When host clusters are deployed in a encryption environment, please follow these recommendations:

- If two encryption engines are part of an HA cluster, configure the host/target pair so they have different paths from both encryption engines. Avoid connecting both the host/target pairs to the same encryption engine. This connectivity does not give the full redundancy needed in case of encryption engine failure and failover to another encryption engine in an HA cluster.
- For Windows-based host clusters, when a quorum disk is used, the quorum disk plays a vital role in keeping the cluster synchronized. It is recommended that you configure the quorum disk to be outside of the encryption environment.
- For AIX-based Power HA System Mirror host clusters, the cluster repository disk should be defined outside of the encryption environment.

HA Cluster deployment considerations and best practices

It is mandatory that the two encryption engines in the HA cluster belong to two different nodes for true redundancy. This is always the case for Brocade Encryption Switches, but is not true if two FS8-18 blades in the same DCX Backbone chassis are configured in the same HA cluster. In Fabric OS v6.3.0 and later releases, HA cluster creation is blocked when encryption engines belonging to FS8-18 blades in the same DCX Backbone chassis are specified.

Key Vault Best Practices

- Make sure that the time difference on the Brocade Encryption Switch and the DPMkey vault does not exceed one minute.
- When encrypted disk LUNs are to be configured or moved to an Encryption Group that uses a different key vault, make sure to decommission the encrypted LUNs from the old Encryption Group.

Tape Device LUN Mapping

When performing LUN mapping, ensure that a given LUN number from a backend physical target is the same across all initiators in the container. Failure to do so can result in unpredictable switch behavior including blade/switch faults. Use the following command to list the LUNs in the target.

```
switch:admin> cryptocfg --discoverLUN <container name>
```

NOTE
It is recommended that you follow the above rule if a given LUN on the backend target is LUN mapped to different initiators.

Chapter

Maintenance and Troubleshooting 6

In this chapter

Encryption group and HA cluster maintenance

This section describes advanced configuration options that you can use to modify existing encryption groups and HA clusters, and to recover from problems with one or more member nodes in the group.

All group-wide configuration commands are executed on the group leader. Commands that clear group-related states from an individual node are executed on the node. The commands require Admin or SecurityAdmin permissions.

Displaying encryption group configuration or status information

You can use the - -**show** -**egstatus** command to display encryption group configuration information and encryption group status information.

- --**show** -**egstatus** -**cfg** Displays encryption group configuration information.
- --**show** -**egstatus** -**stat** Displays encryption group status information.

Removing a member node from an encryption group

This procedure permanently removes a member node from an encryption group, as shown in Figure 108. Upon removal, the HA cluster failover capability and target associations pertaining to the node are no longer present. To remove a node from a group without disrupting these relationships, use the **cryptocfg** --**replace** command. Refer to the section "Replacing an HA cluster member" on page 249 for instructions.

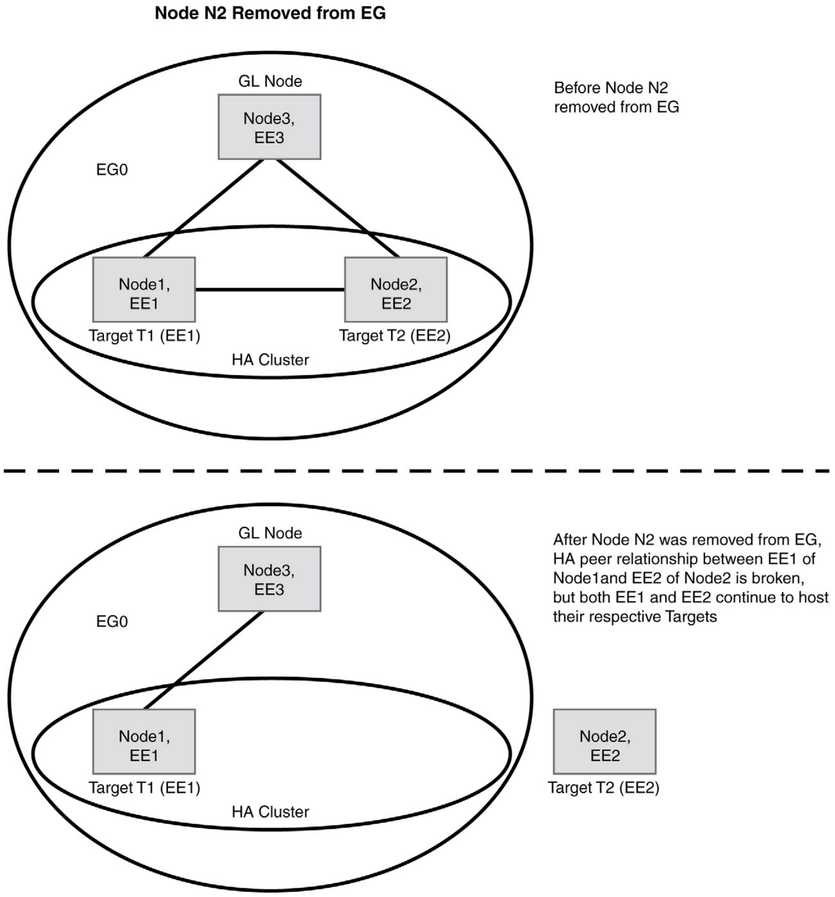

Node N2 Removed from EG
GL Node
Node3, EE3
EG0
Node1, EE1
Node2, EE2
Target T1 (EE1)
Target T2 (EE2)
HA Cluster
Before Node N2 removed from EG
GL Node
Node3, EE3
EG0
Node1, EE1
Target T1 (EE1)
HA Cluster
Node2, EE2
Target T2 (EE2)
After Node N2 was removed from EG, HA peer relationship between EE1 of Node1and EE2 of Node2 is broken, but both EE1 and EE2 continue to host their respective Targets

FIGURE 108 Removing a node from an encryption group

The procedure for removing a node depends on the node's status within an encryption group. HA cluster membership and Crypto LUN configurations must be cleared before you can permanently remove a member node from an encryption group. To remove a node from an encryption group, complete the following steps:

1. Log in to the group leader as Admin or SecurityAdmin.
2. If the node is part of an HA cluster, you must remove it. To do so, complete the following steps:
   a. Remove the node from the HA cluster using the **cryptocfg --rem -haclustermember** command.
   b. Clear all CryptoTarget configurations from the member node using the **cryptocfg --delete -container** command, or move the container using the **cryptocfg --move - container command.**
3. Determine the state of the node. Log in to the member node and enter the **cryptocfg --show -groupmember** command followed by the node WWN. Provide a slot number if the encryption engine is a blade.

```
SecurityAdmin:switch> cryptocfg --show -groupmember \
10:00:00:05:1e:41:99:bc
Node Name:                      10:00:00:05:1e:41:99:bc  (current node)
    State:                      DEF_NODE_STATE_DISCOVERED
    Role:                      MemberNode
    IP Address:                 10.32.33.145
    Certificate:                10.32.33.145_my_cp_cert.pem
    Current Master Key State:   Saved
    Current Master KeyID:
b8:2a:a2:4f:c8:fd:12:e2:a9:25:d9:5b:58:2c:96:7e
    Alternate Master Key State: Not configured
    Alternate Master KeyID:
00:00:00:00:00:00:00:00:00:00:00:00:00:00:00:00

    EE Slot:                    0
        SP state:               Online
        Current Master KeyID:
b8:2a:a2:4f:c8:fd:12:e2:a9:25:d9:5b:58:2c:96:7e
        Alternate Master KeyID:
00:00:00:00:00:00:00:00:00:00:00:00:00:00:00:00
        No HA cluster membership
```

a. If the node is in the DISCOVERED state and the security processor (SP) state is **online** (as shown above), you can remove the node from the encryption group. Complete step 4 and step 5, which completes the procedure.

b. If the node is not in the DISCOVERED state, and you want to remove the node from the encryption group, you must first deregister the node. To do this, log in to the group leader and enter the **cryptocfg --dereg -membernode** command followed by the node WWN.

```
SecurityAdmin:switch> cryptocfg --dereg -membernode 10:00:00:05:1e:41:99:bc
Operation succeeded.
```

4. Reclaim the WWN of the member node.

   a. Enter the **cryptocfg --reclaimWWN -membernode <node-WWN>** command on the group leader to reclaim the VI/VT WWN base for node to be removed.

      When prompted, enter **yes**.

   b. Enter the **cryptocfg --commit** command on the group leader to propagate the change to all nodes in the encryption group:

5. On the group leader, enter the **cryptocfg --eject -membernode** command followed by the node WWN.

```
SecurityAdmin:switch> cryptocfg --eject -membernode 10:00:00
:05:1e:55:3a:f0
WARNING: Before ejecting the membernode, ensure that the VI/VT WWN's
are reclaimed.
Refer to "cryptocfg --reclaimWWN" commands.
ARE YOU SURE  (yes, y, no, n): [no] Node eject granted by protocol clients
[10:00:00:05:1e:55:3a:f0]
Eject node status: Operation Succeeded.
```

6. Deregister the member node to converge the encryption group.

```
SecurityAdmin:switch> cryptocfg --dereg -membernode 10:00:00:05:1e:55:3a:f0
```

7. Log in to the member node and execute the **cryptocfg --reclaimWWN -cleanup** command.

Deleting an encryption group

You can delete an encryption group after removing all member nodes following the procedures described in the previous section. The encryption group is deleted on the group leader after you have removed all member nodes.

Before deleting the encryption group, it is highly recommended that you remove the group leader from the HA cluster and clear all CryptoTarget and tape pool configurations for the group.

The following example deletes the encryption group “brocade”.

1. Log in to the group leader as Admin or SecurityAdmin
2. Enter the **cryptocfg --delete -encgroup** command followed by the encryption group name.

```
SecurityAdmin:switch> cryptocfg --delete -encgroup CRYPTO_LSWAT
This will permanently delete the encryption group configuration
ARE YOU SURE  (yes, y, no, n): [no] y
Encryption group delete status: Operation Succeeded.
```

Removing an HA cluster member

Removing an encryption engine from an HA cluster “breaks” the HA cluster by removing the failover/failback capability for the removed encryption engines, However, the removal of an encryption engine does not affect the relationship between configured containers and the encryption engine that is removed from the HA cluster. The containers still belong to this encryption engine and encryption operations continue.

The remove command should not be used if an encryption engine which failed over exists in the HA Cluster. Refer to the section “Replacing an HA cluster member” on page 249 for instructions on replacing a failed encryption engine in an HA cluster.

1. Log in to the group leader as Admin or SecurityAdmin.
2. Enter the **cryptocfg --remove -haclustermember** command. Specify the HA cluster name and the node WWN to be removed. Provide a slot number if the encryption engine is a blade. The following example removes HA cluster member 10:00:00:05:1e:53:74:87 from the HA cluster HAC2.

```
SecurityAdmin:switch>cryptocfg --remove -haclustermember HAC2 \
10:00:00:05:1e:53:74:87
Remove HA cluster member status: Operation Succeeded.
```

3. Enter **cryptocfg --commit** to commit the transaction.

Displaying the HA cluster configuration

NOTE
The correct failover status of an HA cluster will only be displayed on the HA cluster member nodes in the encryption group.

1. Log in to the group leader as Admin or SecurityAdmin.
2. Enter **the cryptocfg --show -hacluster -all** command.

   In the following example, the encryption group brocade has two HA clusters. HAC 1 is committed and has two members. HAC 2 has one member and remains in a defined state until a second member is added and the transaction is committed.

```
SecurityAdmin:switch>cryptocfg --show -hacluster -all
Encryption Group Name: brocade
Number of HA Clusters: 2

HA cluster name: HAC1 - 2 EE entries
Status:          Committed
         WWN             Slot Number   Status
11:22:33:44:55:66:77:00      0         Online
10:00:00:05:1e:53:74:87      3         Online

HA cluster name: HAC2 - 1 EE entry
Status:          Defined
         WWN             Slot Number   Status
10:00:00:05:1e:53:4c:91      0         Online
```

   In the following example, the encryption group brocade has one HA cluster HAC3. The encryption engine with the WWN of 10:00:00:05:1e:53:89:dd has failed over containers from the encryption engine with the WWN of 10:00:00:05:1e:53:fc:8a it is offline.

```
SecurityAdmin:switch>cryptocfg --show -hacluster -all
Encryption Group Name: brocade
Number of HA Clusters: 1
HA cluster name: HAC3- 2 EE entries
Status: Committed
          WWN            Slot Number     Status
10:00:00:05:1e:53:89:dd      0           Online - Failover active
10:00:00:05:1e:53:fc:8a      0           Offline
```

NOTE
In this particular case, the correct status of **Failover active** is displayed only if group leader node is queried. If the other node is queried, **Failover active** is not displayed, which is not consistent with the actual HA status.

Replacing an HA cluster member

1. Log in to the group leader as Admin or SecurityAdmin.
2. Enter the **cryptocfg --replace -haClusterMember** command. Specify the HA cluster name, the node WWN of the encryption engine to be replaced, and the node WWN of the replacement encryption engine. Provide a slot number if the encryption engine is a blade. The replacement encryption engine must be part of the same encryption group as the encryption engine that is replaced.

```
SecurityAdmin:switch>cryptocfg --replace -haclustermember HAC2 \
10:00:00:05:1e:53:4c:91 10:00:00:05:1e:39:53:67
Replace HA cluster member status: Operation Succeeded.
```

3. Enter **cryptocfg --commit** to commit the transaction.

Case 1: Replacing a failed encryption engine in an HA cluster

Assume a working HA cluster with two operational encryption engines, EE1 and EE2. The target T1 is hosted on EE1 and target T2 is hosted on EE2. Refer to Figure 109.

EE2 fails and generates an offline notification. The target hosted on EE2 (T2 in this case) automatically fails over to EE1. Even though the target T2 is now hosted on EE1 because of the failover process, the target association is still EE2, and the container status is displayed on the hosting node as failover. Use the **cryptocfg --show -container <crypto target container name> -stat** command to display the container status.

1. Invoke the **cryptocfg --replace -haclustermember** command on the group leader to replace the failed encryption engine (EE2) with another encryption engine (EE3). This operation effectively removes the failed encryption engine (EE2) from the HA cluster and adds the replacement encryption engine (EE3) to the HA cluster. The target associations (T2) from the failed encryption engine (EE2) are transferred to the replacement encryption engine (EE3).
2. Commit the transaction. If failback mode is set to **auto**, the target (T2) which failed over earlier to EE1 automatically fails back to the replaced encryption engine (EE3).
3. Invoke the **cryptocfg --reclaimWWN -EE** command on the group leader followed by WWN of the DCX Backbone chassis and the slot number of the encryption engine to be removed.
4. Invoke the **cryptocfg --commit** command to sync the configuration in the encryption group.
5. After the transaction is committed, remove the failed encryption engine from the encryption group.

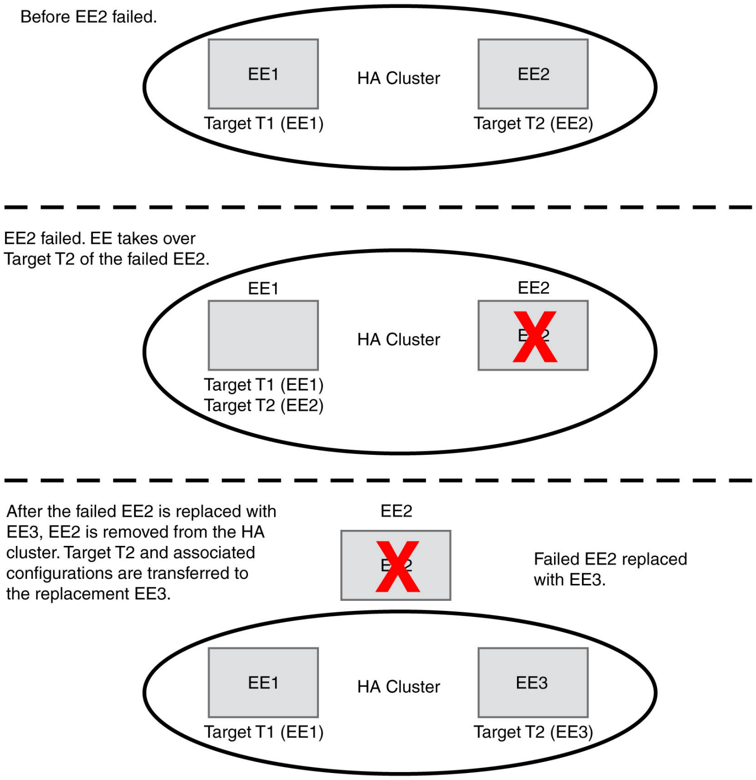


FIGURE 109 Replacing a failed encryption engine in an HA cluster

Case 2: Replacing a “live” encryption engine in an HA cluster

1. Invoke the **cryptocfg --replace -haclustermember** command on the group leader to replace the live encryption engine EE2 with another encryption engine (EE3). This operation effectively removes EE2 from the HA cluster and adds the replacement encryption engine (EE3) to the HA cluster. The target associations (T2) from the replaced encryption engine (EE2) are transferred to the replacement encryption engine (EE3).
2. Commit the transaction.
3. Invoke the **cryptocfg --reclaimWWN -EE** command on the group leader followed by WWN of the DCX Backbone chassis and the slot number of the failed encryption engine.
4. Invoke the **cryptocfg --commit** command to sync the configuration in the encryption group.
5. Remove the encryption engine EE2 from the encryption group

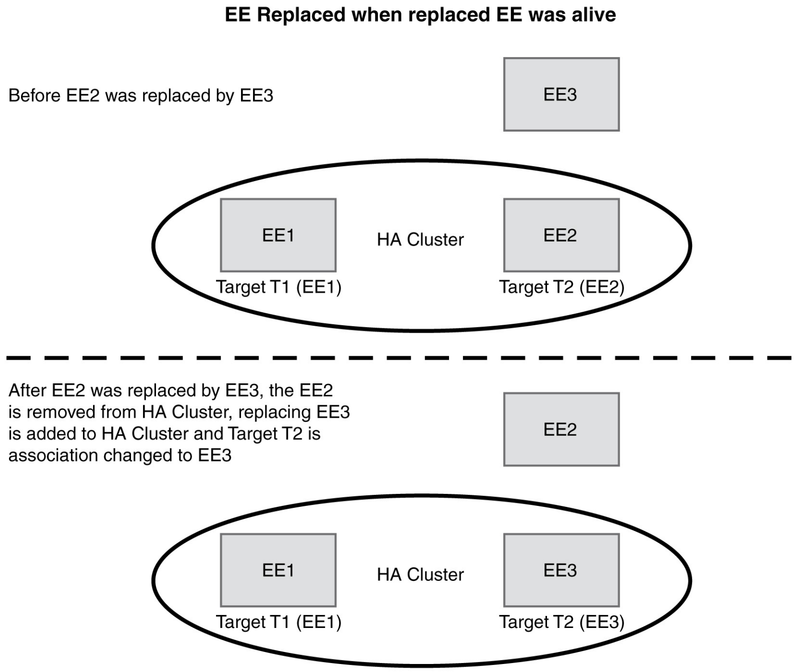

EE Replaced when replaced EE was alive
Before EE2 was replaced by EE3
EE3
EE1
HA Cluster
EE2
Target T1 (EE1)
Target T2 (EE2)
After EE2 was replaced by EE3, the EE2 is removed from HA Cluster, replacing EE3 is added to HA Cluster and Target T2 is association changed to EE3
EE2
EE1
HA Cluster
EE3
Target T1 (EE1)
Target T2 (EE3)

FIGURE 110 Replacing a “live” encryption engine in an HA cluster.

Deleting an HA cluster member

This command dissolves the HA cluster and removes failover capability from the participating encryption engines.

1. Log in to the group leader as Admin or SecurityAdmin.
2. Invoke the **cryptocfg --delete -hacluster** command. Specify the name of the HA cluster you want to delete.

```
SecurityAdmin:switch>cryptocfg --delete -hacluster HAC1
Delete HA cluster status: Operation succeeded.
```

3. Enter the **cryptocfg --commit** command to commit the transaction.

Performing a manual failback of an encryption engine

By default, failback occurs automatically if an encryption engine that failed was replaced or comes back online. When **manual failback** policy is set in the encryption group, you must invoke a manual failback of the encryption engine after the failing encryption engine was restored or replaced. Failback includes all of the encryption engine's target associations. Failback returns all encryption operations to the original encryption engine after it has been restored, or it transfers operations to a replacement encryption engine if the original encryption engine was replaced. The failback operation can only be performed within an HA cluster.

1. Log in to the group leader as Admin or SecurityAdmin.
2. Enter the **cryptocfg --failback -EE** command. Specify the node WWN of the encryption engine to which failover occurred earlier and which is now performing all encryption tasks (current encryption engine), followed by the node WWN of the encryption engine to which failback should occur ("new" encryption engine). Specify a slot number if the encryption engine is a blade.

```
SecurityAdmin:switch>cryptocfg --failback -EE 10:00:00:05:1e:53:4c:91 \
10:00:00:05:1e:39:53:67
Operation Succeeded
```

Failover/failback example

The following example illustrates the states associated with the encryption engines during an active failover and failback process.

- EE2 fails over to EE1.

```
SecurityAdmin:switch> cryptocfg --show -hacluster -all
Encryption Group Name: brocade
Number of HA Clusters: 1

HA cluster name: HAC3- 2 EE entries
Status:          Committed
           WWN                Slot Number   Status
EE1 => 10:00:00:05:1e:53:89:dd        0        Online - Failover active
EE2 => 10:00:00:05:1e:53:fc:8a        0        Offline
```

- The failed EE2 has come back online, Failover is still active:

```
SecurityAdmin:switch> cryptocfg --show -hacluster -all
Encryption Group Name: brocade
Number of HA Clusters: 1

HA cluster name: HAC3 - 2 EE entries
Status:          Committed
           WWN                Slot Number   Status
EE1 => 10:00:00:05:1e:53:89:dd        0        Online - Failover active
EE2 => 10:00:00:05:1e:53:fc:8a        0        Online
```

- A manual failback is issued.

```
SecurityAdmin:switch> cryptocfg --failback -EE 10:00:00:05:1e:53:89:dd 0 \
10:00:00:05:1e:53:fc:8a 0
Operation succeeded.
```

- After the failback completes, the **cryptocfg** **--show** **-hacluster** **-all** command no longer reports active failover.

```
SecurityAdmin:switch> cryptocfg --show -hacluster -all
Encryption Group Name: brocade_1
Number of HA Clusters: 1

HA cluster name: HAC3 - 2 EE entries
Status:          Committed
          WWN                         Slot Number   Status
EE1 => 10:00:00:05:1e:53:89:dd        0       Online
EE2 => 10:00:00:05:1e:53:fc:8a        0       Online
```

Encryption group merge and split use cases

This section describes the following recovery scenarios and related operations:

- "A member node failed and is replaced" on page 253
- "A member node reboots and comes back up" on page 254
- "A member node lost connection to the group leader" on page 255
- "A member node lost connection to all other nodes in the encryption group" on page 255
- "Several member nodes split off from an encryption group" on page 256
- "Adjusting heartbeat signaling values" on page 257
- "EG split possibilities requiring manual recovery" on page 258

A member node failed and is replaced

Assume N1, N2 and N3 form an encryption group and N2 is the group leader node. N3 and N1 are part of an HA cluster. Assume that N3 failed and you want to replace the failed N3 node with an alternate node N4.

Impact

When N3 failed, all devices hosted on the encryption engines of this node failed over to the peer encryption engines in N1, and N1 now performs all of the failed node's encryption services. Rekey sessions owned by the failed encryption engine are failed over to N1.

Recovery

1. Deregister the node N3 from the group leader node.

```
SecurityAdmin:switch> cryptocfg --dereg -membernode <N3 switchWWN>
```

2. Reclaim the WWN base of the failed Brocade Encryption Switch.

```
SecurityAdmin:switch> cryptocfg --reclaim WWN -membernode <N3 switchWWN>
```

3. Synchronize the crypto configurations across all member nodes.

```
SecurityAdmin:switch> cryptocfg --commit
```

NOTE
When attempting to reclaim a failed Brocade Encryption Switch, do not execute **cryptocfg --transabort.** Doing so will cause subsequent reclaim attempts to fail.

4. Set up the member node: Configure the IP address of the new node that is replacing the failed node, and the IP addresses of the I/O cluster sync ports (Ge0 and Ge1), and initialize the node with the **cryptocfg --initnode** command.
5. On the new node that is to be added, invoke **cryptocfg --reclaimWWN -cleanup.**
6. Export the CP certificate from the member node.
7. Import the member node CP certificate into the group leader.
8. On the group leader node, register the member node with the group leader node. Enter the **cryptocfg --reg -membernode** command with appropriate parameters to register the member node. Specify the member node's WWN, Certificate filename, and IP address when executing this command. Successful execution of this command distributes all necessary node authentication data to the other members of the group.

```
SecurityAdmin:switch>cryptocfg --reg -membernode \
10:00:00:05:1e:39:14:00 enc_switch1_cert.pem 10.32.244.60
Operation succeeded.
```

9. Establish the connection between the member node and the key vault.
10. Register the new node with the key manager appliance.
11. On the new node, invoke **cryptocfg --initEE,** and **cryptocfg --regEE** to initialize the encryption engines.
12. After the new node has come online, invoke the **cryptocfg --enableEE** [*slot_number*] command to enable crypto operations on the node's encryption engines.
13. Replace the failed encryption engine on N3 with the encryption engine of the new node N4 to restore broken HA cluster peer relationships. Use the **cryptocfg --replace** command.
14. Remove the failed node from the encryption group. Follow the procedures described in the section "Removing a member node from an encryption group" on page 244.

A member node reboots and comes back up

Assume N1, N2 and N3 form an encryption group and N2 is the group leader node. N3 and N1 are part of an HA cluster. Assume that N3 reboots and comes back up.

Impact

When N3 reboots, all devices hosted on the encryption engines of this node automatically fail over to the peer encryption engine N1. N1 now performs all of N3's encryption services. Any rekey sessions in progress continue. Rekey sessions owned by N3's encryption engine are failed over to N1.

Recovery

If **auto failback** policy is set, no intervention is required. After the node has come back up, all devices and associated configurations and services that failed over earlier to N1 fail back to N3. The node resumes its normal function.

If **auto failback** policy is not set, invoke a manual failback if required. Refer to the section "Performing a manual failback of an encryption engine" on page 252 for instructions.

A member node lost connection to the group leader

AssumeN1, N2 and N3 form an encryption group, and N2 is the group leader node. N3 and N1 are part of an HA cluster. Assume that N3 lost connection to the group leader node N2 but still maintains communications with other nodes in the encryption group.

Impact

Failover to N1 does not occur, because the isolated node and the encryption engines' encryption services continue to function normally. However the disconnect of N3 from the group leader breaks the HA cluster and failover capability between N3 and N1.

You cannot configure any CryptoTargets, LUN policies, tape pools, or security parameters that would require communication with the isolated member node. In addition, you cannot start any rekey operations (auto or manual).

Refer to the section "Configuration impact of encryption group split or node isolation" on page 262 for more information on which configuration changes are allowed.

Recovery

Restore connectivity between the isolated node and the group leader. No further intervention is required.

A member node lost connection to all other nodes in the encryption group

Assume N1, N2 and N3 form an encryption group and N2 is the group leader node. N3 and N1 are part of an HA cluster. Assume that N3 lost connection with all other nodes in the group. Node N3 finds itself isolated from the encryption group and, following the group leader succession protocol, elects itself as group leader. This action splits the encryption group into two encryption group islands. EG1 includes the original encryption group minus the member node N3 that lost connection to the encryption group. EG2 consists of a single node N3, which functions as the group leader. Both EG1 and EG2 are in a degraded state.

Impact

- The two encryption group islands keep functioning independently of each other as far as host I/O encryption traffic is concerned.
- Each encryption group registers the missing members as "offline".

- The isolation of N3 from the group leader breaks the HA cluster and failover capability between N3 and N1.
- You cannot configure any CryptoTargets, LUN policies, tape pools, or security parameters on any of the group leaders. This would require communication with the “offline” member nodes. You cannot start any rekey operations (auto or manual) on any of the nodes. Refer to the section “Configuration impact of encryption group split or node isolation” on page 262 for more information on which configuration changes are allowed.

Recovery

1. Restore connectivity between the two separate encryption group islands.

   When the lost connection is restored, an automatic split recovery process begins. The current group leader and the former group leader (N3 and N2 in this example) arbitrate the recovery, and the group leader with the majority number of members (N2) becomes group leader. If the number of member nodes is the same, the group leader node with the highest WWN becomes group leader.

2. After the encryption group enters the **converged** state, execute the **cryptocfg --commit** command on the group leader node to distribute the crypto-device configuration from the group leader to all member nodes.

Should you decide to remove the isolated node N3, follow the procedures described in the section “Removing a member node from an encryption group” on page 244.

Several member nodes split off from an encryption group

Assume N1, N2, N3, and N4 form an encryption group and N2 is the group leader node. N3 and N1 are part of an HA cluster. Assume that both N3 and N4 lost connection with the encryption group but can still communicate with each other. Following the group leader succession protocol, N3 elects itself as group leader to form a second encryption group with itself and N4 as group members. We now have two encryption groups, EG1 (group leader N2 + N1), and EG2 (group leader N3 + N4).

Impact

- The two encryption groups continue to function independently of each other as far as host I/O encryption traffic is concerned.
- Each encryption group registers the missing members as “offline”.
- The isolation of N3 from the original encryption group breaks the HA cluster and failover capability between N3 and N1.
- You cannot configure any CryptoTargets, LUN policies, tape pools, or security parameters on any of the group leaders. This would require communication with the “offline” member nodes. You cannot start any rekey operations (auto or manual) on any of the nodes. Refer to the section“Configuration impact of encryption group split or node isolation” on page 262 for more information on which configuration changes are allowed.

Recovery

1. Restore the connection between the nodes in the separate encryption group islands, that is, between nodes N3, N4 and between nodes N1 and N2.

   When the lost connection is restored, an automatic split recovery process begins. The two group leaders (N3 and N2 in this example) arbitrate the recovery, and the group leader node with the highest WWN becomes group leader. If the number of nodes in each group is not equal, the group leader for the group with the largest number of members becomes group leader.

2. After the encryption group enters the **converged** state, execute the **cryptocfg** --**commit** command on the group leader node to distribute the crypto-device configuration from the group leader to all member nodes.

Adjusting heartbeat signaling values

Encryption group nodes use heartbeat signaling to communicate to one another and to their associated key vaults. A configurable threshold of heartbeat misses determined how long an encryption group leader will wait before declaring a member node unreachable. The default heartbeat signaling values are three heartbeat misses, each followed by a two second heartbeat time-out. If three consecutive heartbeats are missed (by default, a time interval of six seconds without a heartbeat signal), the encryption group leader node declares a member node as unreachable, resulting in an encryption group split scenario (EG split).

If the management network becomes congested or unreliable resulting in excessive auto-recovery processing or the need for manual recovery from EG splits, it is possible to set larger heartbeat and heartbeat time-out values to mitigate the chances of having the EG split while the network issues are being addressed. The following commands are issued from the encryption group leader nodes to change the heartbeat signaling values.

```
switch:admin> cryptocfg --set -hbmisses <number>
switch:admin> cryptocfg --set -hbtimeout <time>
```

Where:

<number>
Sets the number of heartbeat misses allowed in a node that is part of an encryption group before the node is declared unreachable and the standby takes over. This value is set in conjunction with the time-out value. It must be configured at the group leader node and is distributed to all member nodes in the encryption group. The value entered specifies the number of heartbeat misses. The default value is 3. The range is 3-14 in integer increments only.

<time>
Sets the time-out value for the heartbeat in seconds. This parameter must be configured at the group leader node and is distributed to all member nodes in the encryption group. The value entered specifies the heartbeat time-out in seconds. The default value is 2 seconds. Valid values are integers in the range between 2 and 9.

NOTE
The collective time allowed (the heartbeat time-out value multiplied by the heartbeat misses) cannot exceed 30 seconds (enforced by Fabric OS). The relationship between –**hbmisses** and –**hbtimeout** determines the total amount of time allowed before a node is declared unreachable. If a switch does not sense a heartbeat within the heartbeat timeout value, it is counted as a heartbeat miss. The default values result in a total time of 6 seconds (timeout value of 2 seconds x 3 misses). A total time of 6-10 seconds is recommended. A smaller value might cause a node to be declared unreachable prematurely, while a larger value might result in inefficiency.

EG split possibilities requiring manual recovery

In the event the encryption group (EG) splits and is unable to auto-recover, manual intervention is required to get your encryption group re-converged. It is important to note that while the encryption group is in a split condition, data traffic is NOT impacted in any way. However, EG splits do impact the control plane which means that during an EG split, modification to your encryption group configuration will not be possible.

When an EG split occurs, communications between one or more members of the encryption group is lost, and EG islands form. An EG island is simply a grouping of EG nodes that still have the ability to communicate with one another. As part of the normal recovery procedure, each EG island will select a group leader (GL) node.

Given that you may have up to four nodes per encryption group, an EG split may leave you with any of the following possible EG split combinations:

- **Two node EG split**, resulting in two single node encryption groups. Each node is a group leader node.
- **Three node EG split**, resulting in one of two outcomes:
  - A two node encryption group with a single group leader node, and one single node encryption group where the node is a group leader.
  - Three single Node EGs, each of which is a group leader.
- **Four node EG split**, resulting in one of three outcomes:
  - One three node encryption group with a single group leader, and one single node encryption group where the node is a group leader.
  - A pair of two node encryption groups, with each encryption group having its own group leader.
  - Four single node encryption groups. Each node is a group leader.

EG split manual recovery steps

Regardless of which particular EG Split combination occurs, the recovery procedure is the same.

The following recovery procedures make the following assumptions:

- The networking issues that caused the EG split have been resolved.
- The output of the **cryptocfg --show -groupcfg** command on every EG island shows the EG status as being **DEGRADED.**

NOTE
If one or more EG status displays as CONVERGED contact technical support as the following procedure will not work.

To re-converge the EG, you will need to perform a series of steps. The following is a listing of the basic steps involved - this listing is followed by an example with the details of each step:

1. Confirm that your EG is not in a CONVERGED state.
2. Determine which GL Node will remain the GL Node once the EG is re-converged.

   It is recommended to pick the GL from the largest EG island that exists (i.e. if you EG islands do not all have the same number of members). For example, if you have an EG island with 3 Nodes and another EG island with just 1 Node, pick the GL from the 3 Node EG island.
3. Use the selected EG island's GL Node to deregister every node that is not in a DISCOVERED state.
4. Go to every other EG island and delete the associated EG.

   NOTE
   One additional step is needed here when a four node encryption group splits into a pair of two node encryption groups, with each encryption group having its own group leader. This single special case is addressed in the "Two node EG split manual recovery example".
5. Reregister all Nodes from that were a part of the other EG islands.
6. Verify your EG is reconverged.

Two node EG split manual recovery example

The following example is a case where you have an EG split of a two node encryption group with nodes named Node181 and Node182. Node181 has WWN 10:00:00:00:05:1e:33:33 and Node182 has WWN 10:00:00:05:1e:55:55:55.

1. Perform the **cryptocfg --show -groupcfg** command from every node in your setup. If the EG is split, the Encryption Group state from each node will show up as CLUSTER_STATE_DEGRADED. If some EG Nodes are showing as CLUSTER_STATE_CONVERGED and others as CLUSTER_STATE_DEGRADED then contact technical support. In our case, assume the User has performed this command on both Node181 and Node182 and in each case the result was 'CLUSTER_STATE_DEGRADED'.
2. Determine which node will be encryption group leader when the EG is re-converged. In this example, Node182 is to become the EG Leader for the EG.
3. Deregister every encryption group node not in a DISCOVERED state.

   From the node that you want to be the encryption group leader when the EG is re-converged (Node182 in this example), determine the encryption group state.

   ```
   Node182:admin-> cryptocfg --show -groupcfg
   ```

   The output of this command should show the Encryption Group state as CLUSTER_STATE_DEGRADED.

   Deregister the group member nodes. In this example, this is Node181 as identified by its WWN.

   ```
   Node182:admin-> cryptocfg --dereg -membernode 10:00:00:05:1e:55:33:33
   ```

Display the encryption group state again.

```
Node182:admin-> cryptocfg --show -groupcfg
```

Node182 should now show up with an Encryption Group state of CLUSTER_STATE_CONVERGED.

In this two node example, there is only one other node in the encryption group, and therefore the is only one node to deregister. When you have a 3:1 split or a 2:2 split, issue the following command from the group leader node you are keeping.

```
Switch:admin-> cryptocfg --show -groupmember -all
```

The output of this command will show you every node that was ever a part of this encryption group. Look at State: for all nodes to determine which ones to deregister. Only the nodes with a state of DEF_NODE_STATE_DISCOVERING must be deregistered from the group leader node you are keeping. The example below shows that the node with WWN 10:00:00:05:1e:c1:9a:86 needs to be deregistered.

```
Switch:admin-> cryptocfg --show -groupmember -all

NODE LIST
Total Number of defined nodes:  4
Group Leader Node Name:         10:00:00:05:1e:54:22:44
Encryption Group state:         CLUSTER_STATE_DEGRADED

…. Output truncated…

        Node Name:                      10:00:00:05:1e:c1:9a:86
            State:                      DEF_NODE_STATE_DISCOVERING

…Output truncated…
```

4. Go to every other encryption group island to delete the encryption group.

NOTE
If you have four encryption nodes that have split into a pair of two node encryption groups, refer to "The 2:2 EG split exception" on page 261 for a description of an additional step to take before deleting the encryption group.

In this example, the encryption group island consists only of Node181. From Node181, enter the following command:

```
Node181:admin-> cryptocfg --delete -encgroup <ENTER EG GROUP NAME HERE>
```

This will permanently delete the encryption group configuration

```
ARE YOU SURE  (yes, y, no, n): [no] yes
WARNING: Tape, HA cluster or Container is configured in this node.
Do you want to do EG deletion and retain Tape, HA cluster or Container
configuration
ARE YOU SURE  (yes, y, no, n): [no] yes
Encryption group delete status: Operation Succeeded.
```

If you now perform a cryptocfg --show -groupcfg, you will see that no encryption group on Node181 is defined:

```
Node181:admin-> cryptocfg --show -groupcfg
```

```
Encryption group not defined: Cluster DB and Persistent DB not present, No
Encryption Group Created or Defined.
```

The 2:2 EG split exception

The encryption group deletion procedure may be done directly in every scenario except when there has been a 2:2 split. In that special case, the other encryption group island consists of one group leader and one member node. The group leader node has taken over the group leader role, and has been successful in contacting one member node, placing the member node in a **DEF_NODE_STATE_DISCOVERED** state. Before you can delete the encryption group, you must eject the discovered member node from the group leader node (EGisland2GLNode in the command example that follows). To determine which node is the discovered member node that needs to be ejected, use the following command:

```
EGisland2GLNode:admin-> cryptocfg --show -groupmember -all

NODE LIST
Total Number of defined nodes:  4
Group Leader Node Name:         10:00:00:05:1e:54:22:44
Encryption Group state:         CLUSTER_STATE_DEGRADED

…. Output truncated…

        Node Name:                      10:00:00:05:1e:c1:9b:91
            State:                      DEF_NODE_STATE_DISCOVERED

…Output truncated…
```

Eject the node shown above which is in the DEF_NODE_STATE_DISCOVERED state using the following command:

```
EGisland2GLNode:admin-> cryptocfg --eject -membernode 10:00:00:05:1e:c1:9b:91
```

You can now delete the encryption group from the member node using the **cryptocfg --delete -encgroup** command, and perform a **cryptocfg --show -groupcfg** command to verify that no encryption group is defined on the member node as was done for Node181 in the two node example, as shown near the beginning of step 4.

5. Reregister all nodes from that were a part of the other encryption group islands.

   From Node182, you need to determine the CP certificate name associated with Node181. Use the following command to look for Node182's CP certificate name:

   ```
   Node182:admin-> cryptocfg --show -file -all
   ```

   The output of this command will display a listing of all imported CP certificates. Identify the certificate associated with Node181 and then use it to re-register Node181 as follows:

   ```
   Node182:admin-> cryptocfg --reg -membernode 10:00:00:05:1e:55:33:33 <node181's
   certificate file name> <node181's IP address>
   ```

   Within a minute or two; the encryption group will re-converge.

6. Verify your encryption group is re-converged.

   ```
   Node181:admin-> cryptocfg --show -groupcfg
   Node182:admin-> cryptocfg --show -groupcfg
   ```

   Both nodes will now show a two node CONVERGED EG in which Node182 is the group leader ode and Node181 is a member Node.

The above manual configuration recovery procedure will work nearly identically for all combinations of EG split scenarios. Simply perform the following steps for the other scenarios:

- Pick one EG/EG Leader to be maintained.
- Using that GL Node, deregister all Nodes which are in a DISCOVERING state as determined by the output of the **cryptocfg --show -groupmember -all** command.
- Go to the other EG islands and delete the EGs.
  - In the one case where the other EG has a member node which is in a DISCOVERED state, you will first need to eject that DISCOVERED Node prior to being allowed to delete that other EG.
- From the only remaining EG/EG leader, reregister the previously deregistered Nodes.
- Confirm the EG is converged.

Configuration impact of encryption group split or node isolation

When a node is isolated from the encryption group or the encryption group is split to form separate encryption group islands, the defined or registered node list in the encryption group is not equal to the current active node list, and the encryption group is in a DEGRADED state rather than in a CONVERGED state. Table 7 and Table 8 list configuration changes that are allowed and disallowed under such conditions

TABLE 7 Allowed Configuration Changes

Configuration Type	Allowed configuration changes
Encryption group	• Adding a node to the encryption group • Removing a node from the encryption group • Invoking a node leave command • Deleting an encryption group • Registering a member node (IP address, certificates)
HA cluster	• Removing an encryption engine from an HA cluster • Deleting an HA cluster
Security & key vault	• Initializing a node • Initializing an encryption engine • Re-registering an encryption engine • Zeroizing an encryption engine

TABLE 8 Disallowed Configuration Changes

Configuration Type	Disallowed configuration changes
Security & key vault	• Register or modify key vault settings • Generating a master key • Exporting a master key • Restoring a master key • Enabling or disabling encryption on an encryption engine
HA cluster	• Creating an HA cluster • Adding an encryption engine to an HA cluster • Modifying the failback mode

TABLE 8 Disallowed Configuration Changes

Configuration Type	Disallowed configuration changes
Crypto Device (target/LUN/tape)	• Creating a CryptoTarget container • Adding initiators or LUNs to a CryptoTarget container • Removing initiators or LUNS from a CryptoTarget container • Modifying LUNs or LUN policies • Creating or deleting a tape pool • Modifying a tape pool policy • Starting a manual rekeying session • Performing a manual failback of containers • Deleting a CryptoTarget container

Encryption group database manual operations

Manual intervention may be necessary if the encryption group databases or security databases of encryption group members are not synchronized. The following sections describe manual operations that enable you to do the following:

- Synchronize the encryption group database.
- Synchronize the security database.
- Abort a pending database transaction.

Manually synchronizing the encryption group database

The --**sync** -**encgroup** command manually synchronizes the encryption group database belonging to the group leader node with the databases of all member nodes that are out of sync. If this command is invoked when the encryption group databases are in sync, the command is ignored.

NOTE
When the encryption group is out of sync and the group leader reboots, the newly selected group leader pushes its database information to all other members. The new group leader's database information may be different from what was set up before the group leader was rebooted.

Manually synchronizing the security database

This operation can resolve problems with master key propagation (and connectivity issues between peer node encryption engines in an encryption group). The synchronization occurs every time this command is executed regardless of whether or not the security database was synchronized across all nodes in the encryption group.

Use the --**sync** -**securitydb** command to distribute the security database from the group leader node to all member nodes. This command is valid only on the group leader.

In scenarios where this master key propagation issue still persists, exporting the master key to a file and recovering it resolves the issue. To do this, use the following commands:

- Use the **cryptocfg** --**exportmasterkey** -**file** option to export the master key to a file.
- Use the **cryptocfg** --**recovermasterkey currentMK** -**srcfile** to recover the master key.

Aborting a pending database transaction

You can abort a pending database transaction for any device configurations invoked earlier through the CLI or BNA interfaces by completing the following steps.

1. Use the --**transshow** command to determine the currently pending transaction ID.

   The --**transshow** command displays the pending database transaction for any device configurations invoked earlier through the CLI or BNA interfaces. The command displays the transaction status (completed or pending), the transaction ID, and the transaction owner (CLI or BNA).

2. Use the --**transabort <transaction_ID>** command to abort the transaction, where **<transaction_ID>** specifies the ID of the transaction to be aborted.

Key vault diagnostics

With the introduction of Fabric OS 7.0.0, you can run key vault diagnostics tests to identify any key vault connectivity or key operation errors. You configure the key vault diagnostic test using the **cryptocfg** --**kvdiag** command.

If an encryption switch is part of an EG, the diagnostic testing is performed on that switch only and not the entire group. If multiple nodes in an encryption group have different Fabric OS versions, only those nodes running Fabric OS 7.0.0 and later can be configured for periodic key vault diagnostic testing.

You can set the diagnostic tests to run at regular intervals. When incidents occur, the findings are collected in log reports. The first instance of a failure and subsequent restoration of operation is reported as a Remote Access Server (RAS) log. Subsequent findings for the same incident are not logged to avoid redundant messages.

Key vault connectivity

Key vault connectivity is adiagnostics feature that allows you to periodically collect information about the state of key vault connectivity from the Brocade Encryption Switch and possible version, configuration, or cluster information of the key vault (KV).

This feature reports the following types of configuration information:

- Key Vault/Cluster scope:
  - CA Certificate and its validity (for example, valid header and expiry date)
  - Key Vault IP/Port
  - KV firmware version
  - Time of day on the KV
  - Key class and format on the KV configured for the user group
  - Client session timeout
- Encryption node scope
  - Node KAC certificate and its validity (for example, valid header and expiry date)
  - Username/password
  - User group

- Time of day on the switch
- Key Vault client SDK version
- Timeout and retry policy for the client SDK

The key vault client SDK version, and timeout and retry policy for the client SDK could differ across encryption nodes, depending on the firmware versions they are running.

This feature also reports the results of a vault connectivity check and the results of a validation check on key operations. These results are specific to each encryption node. The operations done as part of this are:

- Connects to the key vault and performs a connectivity check, reports any possible issues in case of failure, for example, certificate issues, username or password issues, or connectivity issues.
- Attempts to retrieve a key and indicates any possible issues in case of failure.
- Attempts to store a key on the vault and indicates any possible issues in case of failure.
- Verifies if a key written is synchronized across the vaults in a cluster.

  This check indicates only the synchronization capability at a given point of time, and does not mean all keys on the vault are synchronized. The need for manual synchronization of keys depends on the point of key vault connectivity failure or user-initiated operations (for example, reboot) and is not identified by the KV diagnostics report. However if such a failure occurs when diagnostics tests are run, failures will be identified and indicated.
- Displays any errors returned from the key vault and indicates the possible issue with configuration or setup that needs manual intervention, such as synchronization of keys or reissuing certificates.
- In a situation whereby a key cannot be created on the vault, (for example, an error message shows “key exists,” “not enough permissions,” or “key creation failure”), verifies the failure and provides additional information. The information shown will vary based on the key vault type.

For additional command information, refer to the *Fabris OS Command Reference v7.0.0.*

Measuring encryption performance

With the introduction of Fabric OS v7.1.0, you can monitor the throughput of redirected I/O flow through an encryption engine (EE). In support of this functionality, the **cryptocfg --perfshow** command is used.

The **cryptocfg --perfshow** command displays the throughput performance between the external ports and the internal cryptographic processing modules, similar to the way that -**portperfshow** displays throughput performance at the external port. Throughput is measured as Bytes/second.

For example:

```
FabricAdmin:switch> cryptocfg --perfshow [slot] [-rx | -tx | -tx -rx]
[-interval <time in seconds>]
```

Whereby:

- **Slot** displays the throughput of redirected I/O flow through the EE in a given slot of the chassis.
- -**tx** displays the transmit throughput of the redirected I/O.
- -**rx** displays the receive throughput of the redirected I/O.

- **-tx -rx** displays the transmit and receive throughputs of the redirected I/O.
- **Interval** represents a numeric value (in seconds) between refreshes.

Examples of the command output are shown below. The port number mentioned is the user port number corresponding to the 8G capable FC platform/port facing towards the Encryption FPGA.

NOTE
For accurate results, ensure that the encryption engines (EEs) are online before executing the command. If an EE is offline, throughput results will be shown as 0.

```
FabricAdmin:switch> cryptocfg --perfshow
 32     33     34     35    36    37    38    39     40    41   42    43
 =====  =====  =====  ===== ====  ====  ====  ====   ====  ==== ====  ====
 5.4m   5.1m   0      0     0     0     5.4m  47.5m  0     0    0     0
 44     45     46     47    48    49    50    51     52    53   54    55    Total
 =====  =====  =====  ===== ====  ====  ====  ====   ====  ==== ====  ====
 0      0      0      0     0     0     0     0      0     0    0     0     75.6m
```

In a DCX Backbone, the slot number is also displayed, along with the performance.

```
dcx:Admin> cryptocfg --perfshow

slot2:
 80     81     82     83    84    85    86    87     88    89   90    91
 =====  =====  =====  ===== ====  ====  ====  ====   ====  ==== ====  ====
 5.4m   5.1m   0      0     0     0     5.4m  47.5m  0     0    0     0
 92     93     94     95    95    97    98    99     100   101  102   103   Total
 =====  =====  =====  ===== ====  ====  ====  ====   ====  ==== ====  ====
 0      0      0      0     0     0     0     0      0     0    0     0     75.6m
```

NOTE
Encryption performance throughput is sensitive to system configuration and will vary. For assistance in optimizing performance, please refer to the *Encryption Best Practices Guide*.

General encryption troubleshooting

Table 9 lists the commands you can use to check the health of your encryption setup. Table 10 provides additional information for failures you might encounter while configuring switches using the CLI.

TABLE 9 General troubleshooting tips using the CLI

Command	Activity
`supportsave`	Check whole system configuration. Run RAS logs. Run RAS traces. Run Security Processor (SP) logs (mainly `kpd.log`).
`configshow`	Check whole system persistent configuration database dump. Check for SPM-, CVLM-, and CNM-related persistent database entries.
`cfgshow`	Check for redirection zones starting with "red_xxx" in defined database for virtual and physical devices.
`nsshow`	Check for crypto virtual target and crypto virtual initiator entries for VT/VI
switch:SecurityAdmin> **`cryptocfg --show -groupcfg`**	Check key vault connection status. Check encryption group/cluster status. **Note:** CONVERGED status means the cluster is formed successfully.
switch:SecurityAdmin> **`cryptocfg --show -groupmember -all`**	1 Check encryption group/cluster member status. **Note:** DISCOVERED state means the member is currently part of a cluster. 2 Check encryption engine/SP and KEK status. **Note:** SP state ONLINE means encryption engine is enabled for encryption with valid KEK (Link Key or Master Key).

TABLE 10 General errors and conditions

Problem	Resolution
Connection to a key vault returns a "Not Responding" message.	Determine if the default port has been changed on the key vault.
After you create an encryption group using RKM/DPM, a newly created container's LUN state changes between "Write metadata is pending" and "Write metadata is in progress" with continuous [RKD-1001] messages displayed on the console.	Power cycle the DCX chassis and then issue the **cryptocfg --enableEE** [*slot number*] command to bring the container's LUN state to Encryption Enabled.If the eth0 IP address on the Brocade Encryption Switch or on the FS8-18 port blade has been modified, a reboot is required.
LUN state for some LUNS remains in "initialize" state on the passive path.	This is expected behavior. The LUNs exposed through Passive paths of the target array will be in either Initialize or LUN Discovery Complete state so long as the paths remain in passive condition. When the passive path becomes active, the LUN changes to Encryption Enabled. Use the **--show -LUN** command with the **-stat** option to check the LUN state.

TABLE 10 General errors and conditions

Problem	Resolution
A backup fails because the LUN is always in the initialize state for the tape container. Tape media is encrypted and gets a key which is archived in the key vault. The key is encrypted with a master key. At a later point in time you generate a new master key. You decide to use this tape media to back up other data. You rewind the tape, erase the tape, relabel the tape, and start a backup from the start of the tape. When the first command comes from the host, the key vault is queried for the tape media based on the media serial number. Since this tape media was used previously, the key is already present in the key vault for this media serial number but this key is encrypted with the old master key and that master key is not present in the switch. You cannot create a new key for this tape media because, per policy, there can be only one key per media.	Use one of two resolutions: • Load the old master key on the switch at an alternate location. The key for the tape media can then be decrypted. • Delete the key for the tape media from the key vault. This forces the switch to create a new key for the tape media. Until you start the backup, the LUN remains in “initialize” state.
“Invalid certificate” error message received when doing a KAC certificate exchange between the Brocade Encryption Switch and a key management system appliance. This error is due to the Brocade Encryption Switch time being ahead of the appliance time.	Use one of two resolutions: • Change the appliance time to match the start period of the KAC certificate. • Change the Brocade Encryption Switch time to synchronize with the appliance time. Upon completion, regenerate the KAC certificate and then do another KAC certificate exchange with the appliance.
“Temporarily out of resources” message received during rekey or first time encryption.	Rekey or first-time encryption sessions are pending due to resource unavailability. A maximum of 10 sessions including rekey (manual or auto) and first time encryption sessions are supported per encryption switch or blade and two sessions per target. The system checks once every hour to determine, if there are any rekey or first time encryption sessions pending. If resources are available, the next session in the queue is processed. There may be up to an hour lag before the next session in the queue is processed. It is therefore recommended that you do not schedule more than 12 rekey or first time encryption sessions.
HA cluster creation fails with error, Create HA cluster status: The IO link IP address of the encryption engine (online) is not configured, even though both the addresses are set and accessible.	The IP addresses for the I/O link ports should be configured before enabling the encryption engine. Failure to do so results in unsuccessful HA Cluster creation. If the IP addresses for these ports were configured after the encryption engine is enabled, reboot the encryption switch or slotpoweroff/slotpoweron the encryption blade to sync up the IP address information to the encryption engine.
Rekeying fails with error “Disabled (Key not in sync)”.	Rekeying was started on a remote encryption engine but the local encryption engine is not capable of starting rekey because the key returned from key vault does not match with the Key ID used by remote encryption engine. You will need to re-enable the LUN after the keys are synced between two key vaults properly using the needs to **cryptocfg −−discoverLUN <Container Name>** command.
cryptocfg −−commit fails with message “Default zone set to all access at one of nodes in EG.”	Default zoning must be set to no access.

TABLE 10 General errors and conditions

Problem	Resolution
Decommissioning an R2 LUN (remote replication LUN) fails with a “Decommission LUN failed because of failure in over-writing metadata” error message.	Check the R2 LUN (remote replication LUN) state. If it is in “Disabled (Data Decommissioning Failed” state, it indicates that the partner R1 (local) LUN was decommissioned with the R1 and R2 LUNs in sync. To decommission the R2 LUN, take the following steps: 1 Split the mirror relationship so each LUN can be independently decommissioned. 2 Write enable the LUN. 3 Decommission the LUN.
When a Brocade 7600 application platform is in the data path, I/O errors may be encountered before reaching the scalability limit of 512 LUNs with 16 outstanding I/Os.	There is no workaround other than reconfiguring so that the 7600 and the encryption switch/blade are not in the same data path.
A performance drop occurs when using DPM on a Microsoft Windows system to back up to a Scalar 500i tape library.	Change the DPM behavior to send one request at a time by adding DWORD “BufferQueueSize” under HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Microsoft Data Protection Manager\Agent, and set the value to 1. Then restart DPM servers: MSDPM, DPMLA, DPMRA.
Continuous “Unrecoverable data decompression failure” error messages are observed on the console on write disabled SRDF remote LUNs following a High Availability Cluster failback.	This is an expected behavior, and you may ignore the messages. If any risk of data corruption is detected, the LUN is disabled, and you are informed and prompted to refresh the Data Encryption Key (DEK).
When attempting to add a LUN to a container, the error message “Commit failed (db propagation).” is returned.	If you are using Fabric OS version 6.3.x, you may be attempting to add a LUN after you have reached the limit of 512 LUNs per initiator in a container. Beginning with Fabric OS version 6.4.0, you will receive an error message that informs you that the maximum limit has been reached.
In an HA cluster after failover, when using the **cryptocfg** --**show** -**hacluster** -**all** command, the failover status displays on one cluster member, but does not display on the other cluster member.	In this particular case, the correct status is displayed when group leader node is queried. If the other node is queried, the status not consistent with the actual HA status. To be sure of the correct status issue the **cryptocfg** --**show** -**hacluster** -**all** command on the group leader node.
You might observe continuous “unrecoverable data decompression failure” (CVLC-1017) messages on Mirror LUNs (R2) when they are WriteDisabled during HA Cluster failback. If you have not performed refreshDEK/port toggle/LUN discovery after the R2 LUN is established from the R1 LUN, host I/Os will result in corruption. I/Os to the metadata region of the R2 LUN will result as “Unrecoverable data decompression failure” RASLOGs.	If you want to write to it, first set the LUNs to RW_enable, then do a discover LUN or refreshDEK, or toggle the target port. Be sure to follow the correct procedures to make the R2 LUN available to the host after it is established from the R1.

Troubleshooting examples using the CLI

Encryption Enabled CryptoTarget LUN

The LUN state should be Encryption enabled for the host to see the Crypto LUN.

```
switch:FabricAdmin> cryptocfg --show -LUN disk_container1 0
21:01:00:e0:8b:a9:ac:d2 -stat

Container name:      disk_container1

Type:                disk

EE node:             10:00:00:05:1e:41:9a:88

EE slot:             0

Target:              50:06:01:60:10:60:06:3a 50:06:01:60:90:60:06:3a

Target PID:          030700

VT:                  20:00:00:05:1e:41:4d:79 20:01:00:05:1e:41:4d:79

VT PID:              012401

Host:                21:01:00:e0:8b:a9:ac:d2 20:01:00:e0:8b:a9:ac:d2

Host PID:            030300

VI:                  20:02:-00:05:1e:41:4d:79 20:03:00:05:1e:41:4d:79

VI PID:              012402

LUN number:          0x0

LUN type:            disk

LUN serial number:   600601604F0B0900847C800FCE0FDD1100000000000000000000E000000
                     000000

Encryption mode:     encrypt

Encryption format:   native

Encrypt existing     disabled
data:

Rekey:               disabled

LUN state:           Encryption enabled

Encryption           AES256-XTS
algorithm:

Key ID state:        Read write

Key ID:              c5:b7:d3:04:53:4b:f8:19:7d:46:87:a7:04:42:68:88

Key creation time:   Thu Jun 26 19:28:27 2008

Operation succeeded
```

Encryption Disabled CryptoTarget LUN

If the LUN state is Encryption Disabled the host will not be able to access the Crypto LUN.

```
switch:FabricAdmin>> cryptocfg --show -LUN disk_container1 0
21:01:00:e0:8b:a9:ac:d2 -stat
Container name:             disk_container1
Type:                       disk
EE node:                    10:00:00:05:1e:43:fe:00
EE slot:                    4
Target:                     50:06:01:61:10:60:06:3a 50:06:01:60:90:60:06:3a
Target PID:                 890c00
VT:                         20:04:00:05:1e:41:4d:79 20:05:00:05:1e:41:4d:79
VT PID:                     01b201
Host:                       21:00:00:e0:8b:89:ac:d2 20:00:00:e0:8b:a9:ac:d2
Host PID:                   890800
VI:                         20:06:-00:05:1e:41:4d:79 20:07:00:05:1e:41:4d:79
VI PID:                     01b202
LUN number:                 0x1
LUN type:                   disk
LUN serial number:          600601604F0B0900857C800FCE0FDD1100000000000000000000
                            F000000000000
Encryption mode:            encrypt
Encryption format:          native
Encrypt existing data:      disabled
Rekey:                      disabled
LUN state:                  Disabled (Unable to retrieve key by key ID found from metadata)
Encryption algorithm:       AES256-XTS
Key ID state:               Read write
Key ID:                     77:93:09:a1:eb:00:af:55:ef:8f:a3:53:e7:a5:9d:ef
Key creation time:          Thu Jun 26 19:28:27 2008
Operation succeeded
```

Management application encryption wizard troubleshooting

Errors related to adding a switch to an existing group

Table 11 lists configuration task errors you might encounter while adding a switch to an existing group, and describes how to troubleshoot them.

TABLE 11 Error recovery instructions for adding a switch to an existing group

Configuration task	Error description	Instructions
Initialize the switch	Unable to add switch to encryption group. The switch is no longer a group leader or does not contain a group.	**Manual option:** To add a switch to the group on the leader switch: 1 Relaunch the **Configure Switch Encryption** wizard and create a new encryption group on the leader switch. 2 When that group is created, launch the **Configure Switch Encryption** wizard again and add the switch to the group.
Initialize the switch	The switch was not properly initialized and was aborted because it is unavailable.	Rerun the **Configure Switch Encryption** wizard for the switch.
Add the switch to the encryption group	Adding the switch to the encryption group failed.	Rerun the **Configure Switch Encryption** wizard for the switch.
Enable the encryption engines	A failure occurred while attempting to enable encryption engines on the switch.	1 Remove the switch from the group using the **Group Members** tab on the **Encryption Group Properties** dialog box. 2 Rerun the **Configure Switch Encryption** wizard for the switch. **Manual Option:** 1 Save the switch's public key certificate to a file using the **Switch Encryption Properties** dialog box. 2 Follow the Key Vault instructions for RSA/Decru/Other key vault.
Save the switch's public key certificate to a file.	The switch's public key certificate could not be saved to a file. **Note:** Verify that the path name and the file name that you are using are both valid and that you have write permissions for the file.	1 Remove the switch from the group using the **Group Members** tab on the **Encryption Group Properties** dialog box. 2 Rerun the **Configure Switch Encryption** wizard for the switch. **Manual Option:** 1 Save the switch's public key certificate to a file using the **Switch Encryption Properties** dialog box. 2 Follow the Key Vault instructions.

Errors related to adding a switch to a new group

Table 12 lists configuration task errors you might encounter while adding a switch to a new group, and describes how to troubleshoot them.

TABLE 12 Error recovery instructions for adding a switch to a new group

Configuration task	Error description	Instructions
Initialize the switch	Unable to initialize the switch due to an error response from the switch.	Diagnose the problem using standard switch CLI commands.
	The switch was not properly initialized and was aborted because it is unavailable.	Rerun the **Configure Switch Encryption** wizard for the switch.
Create encryption group on the switch	A failure occurred while attempting to create a new encryption group on the switch.	1 Click the **Refresh** button on the **Configure Switch Encryption** dialog box to synchronize the data and the database. 2 Rerun the **Configure Switch Encryption** wizard for the switch.
Register one or more key vaults	A failure occurred while attempting to register one or more key vaults for a group on the switch.	1 Remove the switch from the group using the Group Members tab on the **Encryption Group Properties** dialog box. 2 Rerun the **Configure Switch Encryption** wizard for the switch. Manual Option: 1 Launch the **Encryption Group Properties** dialog box and click the **General** tab. 2 From the **General** dialog box, click **Load from File** to install key vault certificates, and then click **OK** to save the information on to the switch. 3 Follow the Key Vault instructions.
Enable the encryption engines	A failure occurred while attempting to enable encryption engines on the switch.	1 Remove the switch from the group using the **Group Members** tab on the **Encryption Group Properties** dialog box. 2 Rerun the **Configure Switch Encryption** wizard for the switch. Manual Option: 1 Launch the **Switch Encryption Properties** dialog box. 2 Save the switch's public key certificate to a file using the **Switch Encryption Properties** dialog box. 3 Follow the Key Vault instructions for the key vault.

TABLE 12 Error recovery instructions for adding a switch to a new group (Continued)

Configuration task	Error description	Instructions
Create a new master key (opaque key vaults only)	A failure occurred while attempting to create a new master key.	1 Remove the switch from the group using the **Group Members** tab on the **Encryption Group Properties** dialog box. 2 Rerun the **Configure Switch Encryption** wizard for the switch. Manual Option: 1 Launch the **Encryption Group Properties** dialog box, and click **Security**. 2 Click the **Master Key Action** button and select **Create New Master Ke**y to generate a new master key.
Save the switch's public key certificate to a file.	The switch's public key certificate could not be saved to a file. **Note:** Verify that the path name and the file name that you are using are both valid and that you have write permissions for the file.	1 Remove the switch from the group using the **Group Members** tab on the **Encryption Group Properties** dialog box. 2 Rerun the **Configure Switch Encryption** wizard for the switch. Manual Option: 1 Save the switch's public key certificate to a file using the **Switch Encryption Properties** dialog box. 2 Follow the Key Vault instructions

General errors related to the Configure Switch Encryption wizard

Table 13 provides additional information for failures you might encounter while configuring switches using the Configure Switch Encryption wizard.

TABLE 13 General errors related to the Configure Switch Encryption wizard

Problem	Resolution
Initialization fails on the encryption engine after the encryption engine is zeroized.	Reboot the switch.
Configuration Commit fails with message “Default zone set to all access at one of nodes in EG.”	Default zoning must be set to no access.

LUN policy troubleshooting

Table 14 may be used as an aid in troubleshooting problems related to LUN policies.

TABLE 14 LUN policy troubleshooting

Case	Reasons for the LUN getting disabled by the encryption switch	Action taken	If you do not need to save the data:	If you need to save the data:
1	The LUN was modified from encrypt policy to cleartext policy but metadata exists.	LUN is disabled. Reason code: Metadata exists but the LUN policy is cleartext.	Issue the **cryptocfg** **--enable** **-LUN** command on one path of the LUN. This erases the metadata on the LUN and the LUN is then enabled with cleartext policy. Issue the **cryptocfg** **--discoverLUN** command on other paths of the LUN in the DEK cluster to enable the LUN.	Modify the LUN back to encrypt policy.
2	The LUN was set up with an encrypt policy and the LUN was encrypted (metadata is present on the LUN), but the DEK for the key ID present in the metadata does not exist in the key vault.	LUN is disabled. Reason code: Metadata exists but the DEK for the key ID from the metadata does not exist.	Modify the LUN policy to cleartext. The subsequent handling is same as in case 1.	Make sure the key vault has the DEK and when the DEK gets restored to the key vault, perform *one* of the following tasks on one of the paths of the LUN to enable the LUN: • Issue the **cryptocfg** **--discoverLUN** command • Remove the LUN from the container and then add it back • Bounce the target port Then issue the **cryptocfg** **--discoverLUN** command on other paths of the LUN in the DEK cluster.
3	The LUN was set up with an encrypt policy and the LUN was encrypted (metadata is present on the LUN), but the current state of the LUN is cleartext instead of encrypted.	LUN is disabled. Reason code: Metadata exists, but the LUN policy is indicated as cleartext.	Modify the LUN policy to cleartext. The subsequent handling is the same as in case 1.	Remove the LUN from the container and then add the LUN back with the LUN state as encrypted, or issue the **cryptocfg** **--enable** **-LUN** command on one of the paths of the LUN which will enable the LUN by using the appropriate key. Then issue the **cryptocfg** **--discoverLUN** command on other paths of the LUN in the DEK cluster to enable the LUN.

Loss of encryption group leader after power outage

When all nodes in an encryption group, HA Cluster, or DEK Cluster are powered down due to catastrophic disaster or power outage to whole data center, and the group leader node either fails to come back up when the other nodes are powered on, or the group leader is kept powered down, the member nodes might lose information and knowledge about the encryption group. If this happens, no crypto operations or commands (except node initialization) are available on the member node after the power-cycle. This condition persists until the group leader back is online.

When a group leader node fails to come back up, the group leader node can be replaced. Two scenarios are considered:

- When encryption group information is *not* lost by member nodes
- When encryption group information is *also* lost by member nodes

Use the following procedure when encryption group information is not lost by the member nodes and one of the member nodes has taken the role of group leader:

1. From the new group leader node, deregister the old group leader node (which has failed) from the encryption group.

```
FabricAdmin:switch> cryptocfg --dereg –membernode <failed GLswitchWWN>
```

2. Reclaim the WWN base of the failed Brocade Encryption Switch.

```
FabricAdmin:switch> cryptocfg --reclaimWWN –membernode <failed GLswitchWWN>
```

3. Synchronize the crypto configurations across all member nodes.

```
FabricAdmin:switch> cryptocfg --commit
```

NOTE
When attempting to reclaim a failed Brocade Encryption Switch, do not execute **cryptocfg --transabort**. Doing so will cause subsequent reclaim attempts to fail.

4. For any containers hosted on the failed group leader node, issue the **cryptocfg --replace** command to change the WWN association of containers from the failed group leader node to the new group leader node (or any other member node in the encryption group) for all containers on the encryption engine.
5. Synchronize the crypto configurations across all member nodes.

```
FabricAdmin:switch> cryptocfg --commit
```

Use the following procedure to replace the failed group leader node with a new node when encryption group information is lost by member nodes:

1. On the new node, perform the switch/node initialization steps as described in Chapter 3.
2. Create an encryption group on the new node with the same encryption group name as before.
3. Use the **configDownload** command to download previously uploaded group leader node and encryption group configuration files to the new node.
4. For any containers hosted on the failed group leader node, issue the **cryptocfg --replace** command to change the WWN association of containers from failed group leader node to the new group leader node for all containers on the encryption engine.

5. Synchronize the crypto configurations across all member nodes.

   ```
   FabricAdmin:switch> cryptocfg --commit
   ```

MPIO and internal LUN states

The Internal LUN State field displayed within the **cryptocfg --show -LUN** command output does not indicate the host-to-storage path status for the displayed LUN, but rather the internal LUN state as known by the given encryption engine. Due to the transparent and embedded nature of this encryption solution, the host-to-storage array LUN path status can only be displayed by using host MPIO software.

For example, assume there are two paths from a host through two encryption switches to a LUN configured within an active/passive storage array. If the LUN is trespassed and the active and passive paths to the LUN are swapped, the host MPIO software will continue to indicate that only one path is active to the LUN, but the Brocade Encryption Switch internal LUN states for both paths will now likely be displayed as Encryption Enabled.

In active/passive storage array environments, for troubleshooting purposes, you may want to update the encryption engine Internal LUN States to match those of their host MPIO path states. You can do this by running the **cryptocfg --discoverLUN <crypto target container name>** command for the encryption engines that own paths to the LUN in question. This command forces a LUN discovery, causing the encryption engine's Internal LUN State to be updated.

Suspension and resumption of rekeying operations

A rekey may be suspended or fail to start for several reasons:

- The LUN goes offline or the encryption switch fails and reboots. Rekey operations are resumed automatically when the target comes back online or the switch comes back up. You cannot abort an in-progress rekey operation.
- An unrecoverable error is encountered on the LUN and the in-progress rekey operation halts. The following LUN errors are considered unrecoverable:

  ```
  SenseKey: 0x3 - Medium Error.
  SenseKey: 0x4 - Hardware Error.
  SenseKey: 0x7 - Data Protect.
  ```

- An unrecoverable error is encountered during the rekey initialization phase. The rekey operation does not begin and a CRITICAL error is logged. All host I/O comes to a halt. All cluster members are notified.
- For any unrecoverable errors that may occur during any other phase of the process, the rekey operation is suspended at that point and a CRITICAL error is logged. All cluster members are notified. Host I/O to all regions of the LUN is halted. Only READ operations are supported for the scratch space region of the LUN used for storing the status block of the rekey operation.

Once all errors have been corrected you have two recovery options:

- Resume the suspended rekey session. All DEK cluster or HA cluster members must be online and reachable for this command to succeed. If successful, this command resumes the rekey sessions from the point where it was interrupted.

1. Enter the **cryptocfg --resume_rekey** command, followed by the CryptoTarget container name, the LUN number and the initiator PWWN.

   ```
   FabricAdmin:switch> cryptocfg --resume_rekey my_disk_tgt 0x0 \
   10:00:00:05:1e:53:37:99
   Operation Succeeded
   ```

2. Check the status of the resumed rekey session.

   ```
   FabricAdmin:switch> cryptocfg --show -rekey -all
   ```

- Read all data off the LUN and write it to another LUN. In this case, you can cancel the rekey session by removing the LUN from its container and force committing the transaction.

  Stop all traffic I/O from the initiators accessing the LUN before removing the LUN to avoid I/O failure between the initiators and the LUN. If the LUN is exposed to more than one initiator under different LUN Numbers, remove all exposed LUN Numbers. When there are multiple paths to a LUN, you must remove the LUNs from all exposed CryptoTarget containers before you commit the transaction. Failure to do so may result in a potentially catastrophic situation where one path ends up being exposed through the encryption switch and another path has direct access to the device from a host outside the protected realm of the encryption platform.

  1. Enter the **cryptocfg --remove -LUN** command followed by the CryptoTarget container name, the LUN Number, and the initiator PWWN.

     ```
     FabricAdmin:switch> cryptocfg --remove -LUN my_disk_tgt 0x0
     10:00:00:00:c9:2b:c9:3a
     Operation Succeeded
     ```

  2. Commit the configuration with the **-force** option to completely remove the LUN and all associated configuration data in the configuration database. The data remains on the removed LUN in an encrypted state.

     ```
     FabricAdmin:switch> cryptocfg --commit -force
     Operation Succeeded
     ```

FS8-18 blade removal and replacement

The following procedures identify steps for removing and replacing an FS8-18 blade in a DCX Backbone chassis. The procedures assume that the replacement blade is being inserted in the same slot as the old blade that was removed.

- For a multi-node replacement, refer to "Multi-node EG replacement" on page 278.
- For a single-node replacement, refer to "Single-node EG replacement" on page 280.

Multi-node EG replacement

1. Power off the FS8-18 blade. Remove the I/O links and FC cables from the blade, noting where each was attached so the replacement blade can be cabled properly.
2. Log in as Admin or FabricAdmin.

3. If the replaced FS8-18 blade is in member node, invoke the following command to reclaim the base WWN.

   ```
   FabricAdmin:switch> cryptocfg --reclaimWWN –EE <failed EE WWN> <slot number>
   ```

4. Issue commit.

   ```
   FabricAdmin:switch> cryptocfg --commit
   ```

5. Replace the old FS8-18 blade with the new FS8-18 blade and reconnect the FC cables and I/O Link cables.
6. Insert the new FS8-18 blade in the same slot of the chassis that was used by the old FS8-18 blade. Reconnect the I/O sync ports to the same private LAN as the I/O sync ports of the old blade, and confirm that the IP address of the I/O sync ports (Ge0 and Ge1) is same as the previous IP address.
7. Zeroize the new encryption engine (EE) using the following command:

   ```
   FabricAdmin:switch> cryptocfg —zeroizeEE [slotnumber]
   ```

8. Invoke **slotpoweroff** and **slotpoweron** commands.

   ```
   FabricAdmin:switch> slotpoweroff [slotnumber]
   FabricAdmin:switch> slotpoweron [slotnumber]
   ```

9. If the encryption group (EG) has a system card authentication enabled, you need to reregister the system card through the BNA client for the new EE. Refer to Chapter 2, Configuring Encryption Using the Management Application."
10. Initialize the new EE using the following command:

    ```
    FabricAdmin:switch> cryptocfg —initEE [slotnumber]
    ```

11. Register the new EE using the following command:

    ```
    FabricAdmin:switch> cryptocfg --regEE [slotnumber]
    ```

12. Enable the new EE using the following command:

    ```
    FabricAdmin:switch> cryptocfg —enableEE [slotnumber]
    ```

13. Verify the new FS8-18 blade EE has the same master key as the other EEs in the EG using the following command:

    ```
    FabricAdmin:switch> cryptocfg --show –groupmember –all
    ```

14. If a master key is not present, restore the master key from a backed up copy. Procedures will differ depending on the backup media used (for example, recovery smart cards, from the key vault, from a file on the network, or a file on a USB-attached device). Refer to Chapter 2, Configuring Encryption Using the Management Application."
15. Check the EE state using the following command to ensure that the EE is online.

    ```
    FabricAdmin:switch> cryptocfg —show –localEE
    ```

NOTE
Because the FS8-18 blade was inserted in the same slot as the previous blade, no change of HA cluster container ownership is required; the HA cluster configuration is retained.

16. If "manual" failback was set on the HA cluster, you must manually fail back the LUNs owned by the newly replaced EE.
17. Check the EG state using the following command to ensure that the entire EG is in a converged and In Sync state.

```
FabricAdmin:switch> cryptocfg --show -groupcfg
```

Single-node EG replacement

The following procedure applies to single or multiple FS8-18 blades.

1. Power off the FS8-18 blade. Remove the I/O Link and FC cables from the blade, noting where each was attached so the replacement blade can be cabled properly.
2. Replace the old FS8-18 blade with the new FS8-18 blade and reconnect the FC cables and I/O Link cables.
3. Insert the new FS8-18 blade in the same slot of the chassis that was used by the old FS8-18 blade. Reconnect the I/O sync ports to the same private LAN as the I/O sync ports of the old blade, and confirm that the IP address of the I/O sync ports (Ge0 and Ge1) is same as the previous IP address.
4. Log in as Admin or FabricAdmin on the replacement node.
5. Zeroize the new encryption engine (EE) using the following command:

```
FabricAdmin:switch> cryptocfg --zeroizeEE [slotnumber]
```

6. If the encryption group (EG) has a system card authentication enabled, you need to reregister the system card through the BNA client for the new EE. Refer to Chapter 2, Configuring Encryption Using the Management Application."
7. Initialize the new EE using the following command:

```
FabricAdmin:switch> cryptocfg --initEE [slotnumber]
```

8. Register the new EE using the following command:

```
FabricAdmin:switch> cryptocfg --regEE [slotnumber]
```

9. Enable the new EE using the following command:

```
FabricAdmin:switch> cryptocfg --enableEE [slotnumber]
```

10. Verify the new FS8-18 blade EE has the same master key as the other EEs in the EG using the following command:

```
FabricAdmin:switch> cryptocfg --show -groupmember -all
```

11. If a master key is not present, restore the master key from a backed up copy. Procedures will differ depending on the backup media used (for example, recovery smart cards, from the key vault, from a file on the network, or a file on a USB-attached device). Refer to Chapter 2, Configuring Encryption Using the Management Application."

12. Check the EE state using the following command to ensure the EE is online.

```
FabricAdmin:switch> cryptocfg --show -localEE
```

NOTE
Because the FS8-18 blade was inserted in the same slot as the previous blade, no change of HA cluster container ownership is required; the HA cluster configuration is retained.

13. If "manual" failback was set on the HA cluster, you must manually fail back the LUNs owned by the newly replaced EE.

Brocade Encryption Switch removal and replacement

The following procedures identify steps for removing and replacing a Brocade Encryption Switch.

- For a multi-node replacement, refer to "Multi-node EG Case" on page 281.
- For a single-node replacement, refer to "Single-node EG Replacement" on page 284.

Multi-node EG Case

1. If possible, upload the configuration from the group leader node using the Fabric OS **configupload** command.

2. Power off the Brocade Encryption Switch. Remove the Mgmt Link, I/O links, and FC cables from the Brocade Encryption Switch, noting where each was attached so that the replacement Brocade Encryption Switch can be cabled properly.

3. From the group leader node, invoke the following command to deregister the old Brocade Encryption Switch.

```
Admin:switch> cryptocfg --dereg -membernode <WWN of old Brocade Encryption 
Switch>
```

4. From the group leader node, invoke the following command to reclaim the WWN base from the old Brocade Encryption Switch.

```
Admin:switch> cryptocfg --reclaim -membernode <WWN of old Brocade Encryption 
Switch>
```

5. Issue commit.

```
Admin:switch> cryptocfg --commit
```

6. Replace the old Brocade Encryption Switch with the new Brocade Encryption Switch and reconnect the Mgmt link, I/O links, and FC cables.

7. Reconnect the I/O sync ports to the same private LAN as the I/O sync ports of the failed node.

8. Power on the new Brocade Encryption Switch. Note that the FC cables have not yet been plugged in.
9. Set the IP address for the new Brocade Encryption Switch using the ipAddrSet command for the Mgmt and I/O links. Check that the switch name and domain ID associated with the replacement switch match that of the original.
10. Zeroize the new Brocade Encryption Switch using the following command.

```
Admin:switch> cryptocfg --zeroizeEE
```

11. If the encryption group (EG) has a system card authentication enabled, you need to reregister the system card through the BNA client for the new EE. Refer to Chapter 2, Configuring Encryption Using the Management Application."
12. Initialize the new Brocade Encryption Switch node using following command.

```
Admin:switch> cryptocfg --initnode
```

13. Initialize the new EE using the following command.

```
Admin:switch> cryptocfg --initEE
```

14. Register the new EE using the following command.

```
Admin:switch> cryptocfg --regEE
```

15. Enable the new EE using the following command.

```
Admin:switch> cryptocfg --enableEE
```

16. Invoke the following command to clean up the WWN base on the new Brocade Encryption Switch if it was used earlier.

```
Admin:switch> cryptocfg --reclaim -cleanup
```

17. From the new Brocade Encryption Switch node, invoke the following command to export the CP certificate of the new Brocade Encryption Switch.

```
Admin:switch> cryptocfg --export -scp -CPcert <host IP> <host user> <host file
path>
```

18. From the group leader node, invoke the following command to import the new Brocade Encryption Switch node certificate on the group leader node.

```
Admin:switch> cryptocfg --import -scp <Certificate file name> <host IP> <host
user> <host file path>
```

19. From the group leader node, run the following command to register the new Brocade Encryption Switch node as a member node on the group leader.

```
Admin:switch> cryptocfg --reg -membernode <New BES WWN>  <Cert file Name> <Old
IP address>
```

20. Export the KAC CSR from the new node and sign the CSR from the CA that signed the failed node CSR.

21. Import the signed CSR/Cert onto the new node.
22. Register back the signed KAC CSR/Cert onto the new node using the following command.

    ```
    Admin:switch> cryptocfg --reg –KACcert
    ```

23. Remove the existing identity of the failed node from the DPM appliance.
24. Create an identity for the new node, and upload the new node KAC certificate to the DPM appliance.
25. Check the EE state using the following command to ensure that the EE is online.

    ```
    Admin:switch> cryptocfg --show –localEE
    ```

26. From the new Brocade Encryption Switch, invoke the following command to set the default zone as allAccess so the configuration from the existing Fabric is pushed to the new Brocade Encryption Switch.

    ```
    Admin:switch> defzone –allaccess
    ```

27. Invoke the following command on the new Brocade Encryption Switch.

    ```
    Admin:switch> cfgsave
    ```

28. Replace the FC Cables to the new Brocade Encryption Switch.
29. Invoke the cfgsave command on any switch in that fabric. The fabric configuration from the existing fabric will be merged into the new Brocade Encryption Switch.
30. Verify that defzone is set as no access.
31. If HA cluster membership for the old Brocade Encryption Switch was in place, move container movement to the new Brocade Encryption Switch using the following procedure.
    a. Replace the old EE with the new EE using the following command on the group leader.

    ```
    Admin:switch> cryptocfg --replace <WWN of Old BES> <WWN of new BES>
    ```

    b. Issue commit.

    ```
    Admin:switch> cryptocfg --commit
    ```

    c. Replace the HA cluster membership from the old EE to the new EE using the following command on the group leader.

    ```
    Admin:switch> cryptocfg --replace –haclustermember <HA cluster name> <WWN
    of old Brocade Encryption Switch> <WWN of new Brocade Encryption Switch>
    ```

    d. Issue commit.

    ```
    Admin:switch> cryptocfg --commit
    ```

    e. If “manual” failback was set on the HA cluster, user intervention will be required to manually fail back the LUNs owned by the newly replaced Brocade Encryption Switch.

32. If HA cluster membership for the old Brocade Encryption Switch was *not* in place, move container movement to the new Brocade Encryption Switch using the following procedure.

    a. Replace the old EE with the new EE using following command on the group leader.

    ```
    Admin:switch> cryptocfg --replace <WWN of old Brocade Encryption Switch>
    <WWN of new Brocade Encryption Switch>
    ```

    b. Issue commit.

    ```
    Admin:switch> cryptocfg --commit
    ```

33. Check the EG state using the following command to ensure that the entire EG is in the converged and In Sync state.

    ```
    Admin:switch> cryptocfg –-show –groupcfg
    ```

Single-node EG Replacement

1. Upload the configuration stored on the Brocade Encryption Switch you are replacing using the FOS **configupload** command.
2. Power off the Brocade Encryption Switch. Remove the Mgmt Link, I/O links, and FC cables from the Brocade Encryption Switch, noting where each was attached so that the replacement Brocade Encryption Switch can be cabled properly.
3. Power on the new Brocade Encryption Switch. Note that the FC cables have not yet been plugged in.
4. Set the IP address for the new Brocade Encryption Switch using the **ipAddrSet** command for both Mgmt and I/O links. Check that the switch name and domain ID associated with the replacement switch match that of the original.
5. Initialize the new Brocade Encryption Switch node using following command:

    ```
    Admin:switch> cryptocfg --initnode
    ```

6. Zeroize the new Brocade Encryption Switch.

    ```
    Admin:switch> cryptocfg --zeroizeEE
    ```

7. If system card authentication was enabled, you must re-register the system card through the BNA client for the new encryption engine.
8. Initialize the new encryption engine using the following command.

    ```
    Admin:switch> cryptocfg --initEE [slotnumber]
    ```

9. Register the new encryption engine using the following command.

    ```
    Admin:switch> cryptocfg --regEE [slotnumber]
    ```

10. Enable the new encryption engine using the following command.

    ```
    Admin:switch> cryptocfg --enableEE [slotnumber]
    ```

11. Invoke the following command to cleanup any WWN entries which are used earlier.

    ```
    Admin:switch> cryptocfg --reclaim -cleanup
    ```

12. Recreate the EG with the same name as before using the following command.

    ```
    Admin:switch> cryptocfg --create –encgroup <EG name>
    ```

13. Invoke **configdownload** from the previous uploaded configuration.
14. Enable the switch using the **switchenable** command.
15. Deregister both key vaults using the following command.

    ```
    Admin:switch> crypocfg --dereg –keyvault <label name>
    ```

16. Export the KAC CSR from new node and sign the CSR from the CA that signed the failed node CSR.
17. Submit the CSR to a CA.
18. Import the signed KAC certificate onto the new node using the **cryptocfg** `--`**import** command.
19. Register back the signed KAC CSR/Certificate onto the new node using the following command:

    ```
    Admin:switch> cryptocfg --reg -KACcert
    ```

20. Register the new node KAC Certificate with the DPM appliances and create an identity for this node on the DPM appliance in the Identity Group and associate the identity with the newly created signed certificate.
21. Register the DPM appliance cluster virtual IP and CA certificate onto this node. Import the key vault certificate file using the following command:

    ```
    Admin:switch> cryptocfg --import -scp
    ```

22. The DPM appliance can then be registered using the imported file using the following command:

    ```
    Admin:switch> cryptocfg --reg -keyvault
    ```

23. Remove the existing identity of the failed node from the DPM appliance Identity Group.
24. If a master key is not present, restore the master key from a backed up copy. Procedures will differ depending on the backup media used (for example, recovery smart cards, from the key vault, from a file on the network, or a file on a USB-attached device). Refer to Chapter 2, “Configuring Encryption Using the Management Application.”
25. Check the encryption engine (EE) state using following command to ensure that the encryption engine is online.

    ```
    Admin:switch> cryptocfg --show -localEE
    ```

26. Set the defzone as **allAccess** on the new Brocade Encryption Switch, so the configuration from the Fabric is pushed to new Brocade Encryption Switch.

27. Invoke the following command on the new Brocade Encryption Switch:

    ```
    Admin:switch> cfgsave
    ```

28. Reconnect the FC Cables to the new Brocade Encryption Switch.
29. Invoke the **cfgsave** command on any switch in that fabric. The fabric configuration from the existing fabric is merged into the new Brocade Encryption Switch.
30. Verify that **defzone** is set as **no access**.
31. If HA cluster membership for the old Brocade Encryption Switch was in place. Do the following for moving container movement to the new Brocade Encryption Switch.
    a. Replace the old EE with the new EE using the following command on the group leader.

    ```
    Admin:switch> cryptocfg --replace <WWN of Old BES> <WWN of new BES>
    ```

    b. Issue commit.

    ```
    Admin:switch> cryptocfg --commit
    ```

    c. Replace the HAC membership from the old EE to the new EE using the following command on the group leader.

    ```
    Admin:switch> cryptocfg --replace –haclustermember <HA cluster name> <WWN
    of Old BES> <WWN of New BES>
    ```

    d. Issue commit.

    ```
    Admin:switch> cryptocfg --commit
    ```

    e. If “manual” failback was set on the HA cluster, you must manually fail back the LUNs owned by the newly replaced Brocade Encryption Switch.
32. If HA cluster membership for the old Brocade Encryption Switch was *not* in place. Do the following for moving container to the New BES.
    a. Replace the old EE with the new EE using following command on the group leader.

    ```
    Admin:switch> cryptocfg --replace <WWN of Old BES> <WWN of new BES>
    ```

    b. Issue commit.

    ```
    Admin:switch> cryptocfg --commit
    ```

33. Check the EG state using the following command to ensure that the entire EG is in a converged and In Sync state.

    ```
    Admin:switch> cryptocfg –-show –groupcfg
    ```

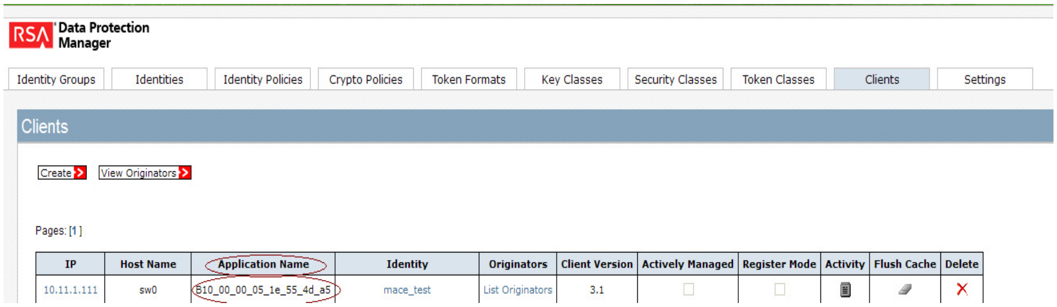
Deregistering a DPM key vault

Each Brocade Encryption Switch is associated with an identity and a client on the DPM 3.2 server. Before reregistering the DPM server on the Brocade Encryption Switch, make sure the previous client entry is removed from the DPM server.

You can identify the client name of the Brocade Encryption Switch on the DPM Key Vault using the **cryptocfg --show -groupcfg** command, which displays the Client Username. A sample output is provided.

```
SecurityAdmin:switch> cryptocfg --show -groupcfg

    Primary Key Vault:
    IP address:        10.11.1.111    Certificate ID:  RSA
    Certificate label:  dpm
    State:            Connected
    Type:             DPM

    Secondary Key Vault not configured

    Additional Key Vault/Cluster Information:
    Key Vault/CA Certificate Validity:      Yes
    Port for Key Vault Connection:          443
    Time of Day on Key Server:              N/A
    Server SDK Version:                     N/A

    Encryption Node (Key Vault Client) Information:
    Node KAC Certificate Validity:          Yes
    Time of Day on the Switch:              N/A
    Client SDK Version:                     RKM-Client  3.1 27-Jan-2012
    Client Username:                        B10_00_00_05_1e_55_4d_a5
    Client Usergroup:                       N/A
    Connection Timeout:                     3 seconds
    Response Timeout:                       25 seconds
    Connection Idle Timeout:                N/A
```

Once identified, the client on the DPM Key Vault with the corresponding name should be deleted when the DPM is deregistered on the Brocade Encryption Switch. Otherwise reregistration of the DPM Key Vault will result in key vault connectivity failure on the Brocade Encryption Switch. (Refer to Figure 111.)



IP	Host Name	Application Name	Identity	Originators	Client Version	Actively Managed	Register Mode	Activity	Flush Cache	Delete
10.11.1.111	sw0	B10_00_00_05_1e_55_4d_a5	mace_test	List Originators	3.1	□	□	▤		✗

FIGURE 111 DPM Clients page

Reclaiming the WWN base of a failed Brocade Encryption Switch

When a Brocade Encryption Switch fails, to reclaim the WWN base, follow these steps:

1. Locate the Brocade Encryption Switch that has failed and deregister from the encryption group.

   ```
   Admin:switch> cryptocfg --dereg –membernode <switchWWN>
   ```

2. Reclaim the WWN base of the failed Brocade Encryption Switch.

   ```
   Admin:switch> cryptocfg --reclaimWWN –membernode <switchWWN> [-list]
   ```

3. Synchronize the crypto configurations across all member nodes.

   ```
   Admin:switch> cryptocfg --commit
   ```

NOTE
When attempting to reclaim a failed Brocade Encryption Switch, do not execute **cryptocfg --transabort.** Doing so will cause subsequent reclaim attempts to fail.

Removing stale rekey information for a LUN

To clean up stale rekey information for a LUN, complete one of the following procedures:

Procedure 1:

1. Modify the LUN policy from “encrypt” to “cleartext” and commit. The LUN will become disabled.
2. Enable the LUN using the following command:

   ```
   Admin:switch> cryptocfg --enable –LUN
   ```

2. Modify the LUN policy from “cleartext” to “encrypt” with the **enable_encexistingdata** command to enable the first-time encryption, then commit. This will clear the stale rekey metadata on the LUN and the LUN can be used again for encryption.

Procedure 2:

1. Remove the LUN from the CryptoTarget Container and commit.

2. Add the LUN back to the CryptoTarget Container with LUN State=”clear-text”, policy=”encrypt” and “enable_encexistingdata” set for enabling the first-time encryption, then commit. This will clear the stale rekey metadata on the LUN and the LUN can be used again for encryption.

Downgrading firmware from Fabric OS 7.1.0

NOTE
When disabling the firmware consistency check, there should be no LUNs with pending decommission or in a failed state. If the firmware download to a version earlier than Fabric OS 7.1.0 is disallowed because of any LUNs under decommission or in a failed state, you must either complete decommissioning, or remove the offending LUNs before retrying **cryptocfg --delete –decommissionedkeyids** to disable the firmware consistency check.

NOTE
You should not join a Fabric OS 7.0.1(x) node into an encryption group or eject a node with Fabric OS 7.1.0 or later when the firmware consistency check for the device decommission feature is enabled in the encryption group.

NOTE
If you are attempting to download firmware to a Fabric OS version earlier than v6.4.0, for example, v6.3.0(x), you might be prompted with the following error message, even if there are no failed decommissioned LUNs, and even if no decommissioned key ID list exists on a node:

"Downgrade is not allowed for this key vault type, as device decommission feature is in use. Please use **cryptocfg --delete –decommissionedkeyids** to disable device decommission. Make sure that no LUN is undergoing decommission or is in failed state.”

If a device decommission firmware consistency check is enabled in the encryption group, firmware downgrades to a Fabric OS version earlier than v6.4. will be blocked until the firmware consistency check for device decommission feature is disabled.

The firmware consistency check for device decommission is *enabled* when you execute the following:

```
SecurityAdmin:switch> cryptocfg --decommission -container <container name>
-initiator <initiatator PWWN> -LUN <lun number>
```

The firmware consistency check for device decommission is *disabled* when you execute the following:

```
SecurityAdmin:switch> cryptocfg --delete –decommissionedkeyids
```

The success of the operation does not mandate that the firmware consistency check be disabled for device decommission.

NOTE
When disabling the firmware consistency check, there should be no LUNs with pending decommission or in a failed state. If the firmware download to a version earlier than Fabric OS 6.4.0 is disallowed because of any LUNs under decommission or in a failed state, you must either complete decommissioning or remove the offending LUNs before retrying **cryptocfg --delete –decommissionedkeyids** to disable the firmware consistency check.

NOTE
You should not join a Fabric OS 6.3.0(x) node into an encryption group or eject a node with Fabric OS 6.4.0 and later when the firmware consistency check for the device decommission feature is enabled in the encryption group.

Fabric OS and DPM Compatibility Matrix

DPM 3.1 introduces the GKA feature, which is incompatible with the RKM 2.1.1 client. DPM 3.2 offers a solution to resolve this incompatiability issue with the RKM 2.1.1 client. DPM 3.1 client is compatible with DPM 3.x servers, but is not compatible with RKM 2.x servers.

Because of the limitations associated with the RKM 2.1.1 client and the DPM 3.1 server, it is recommended that you move to DPM 3.2 server instead of v3.1.

Table 15 identifies Fabric OS and DPM compatibility.

TABLE 15 Compatibility Matrix

Fabric OS version	Client SDK	Server version	
		RKM 2.7.x	**DPM 3.2**
v7.0.1 and earlier	RKM 2.1.1	Yes	Yes
v7.1.0 and later	DPM 3.1	No	Yes

Splitting an encryption group into two encryption groups

In this example, which is represented in Table 16, you have one encryption group with four nodes from which you want to remove two of the nodes and add them to a new encryption group.

TABLE 16 Splitting an encryption group

Encryption group	Nodes
Original EG	FOS1 (Group Leader) FOS2 FOS3 FOS4
New EG1	FOS1 (Group Leader) FOS2
New EG2	FOS3 (Group Leader) FOS4

1. Enter the following command on FOS1 to reclaim the VI/VT WWN base for FOS3:

   ```
   Admin:switch> cryptocfg --reclaimWWN -membernode <FOS3-WWN>
   ```

   When prompted, enter **yes**.

2. Enter the following command on FOS1 to propagate the change to all nodes in the EG:

   ```
   Admin:switch> cryptocfg --commit
   ```

3. Enter the following command in FOS1 to eject node FOS3 from the EG:

   ```
   Admin:switch> cryptocfg --eject -membernode <FOS3-WWN>
   ```

4. Enter the following command on FOS1 to deregister the ejected node from the encryption group:

```
Admin:switch> cryptocfg --dereg -membernode <FOS3-WWN>
```

5. Enter the following command on FOS3 to clean up the encryption configuration on the deregistered node:

   ```
   Admin:switch> cryptocfg --reclaimWWN -cleanup
   ```

   When prompted, enter **yes** to each prompt.

6. Repeat steps 1–5 for FOS4.

7. Create a new EG on FOS3:

   a. Create the group:

   ```
   Admin:switch> cryptocfg --create -encgroup FOS3
   ```

   b. Set the key vault type.

   ```
   Admin:switch> cryptocfg --set -keyvault RKM/DPM
   ```

   When prompted, enter **yes** to each prompt.

8. Add FOS4 as a member node to the new EG.

- For details about adding member nodes to an EG, see“Adding a member node to an encryption group” on page 143.
- For details about creating encryption groups, see “Creating an encryption group” on page 35.

Moving an encryption blade from one EG to another in the same fabric

In this example, which is represented in Table 17, you have two EGs, each containing two nodes. You want to move the blade currently located in DCX1, slot 4 to DCX2, slot 3 in EG2.

TABLE 17 Moving a blade from one EG to another EG

Encryption group	Nodes (before move)	Nodes (after move)
EG1	FOS1 (Group Leader) DCX1—Contains 2 FS-18 blades (slot 2 and slot 4)	FOS1 (Group Leader) DCX1—Contains 1 FS-18 blade in slot 2
EG2	FOS2 (Group Leader) DCX2—Contains 1 FS-18 blade in slot 2	FOS2 (Group Leader) DCX2—Contains 2 FS-18 blades (slot 2 and slot 3)

1. Enter the following command on FOS1 to reclaim the VI/VT WWN base for the encryption engine to be moved out of the EG.

   ```
   Admin:switch> cryptocfg --reclaimWWN -EE <DCX1_WWN> 4
   ```

   When prompted, answer **yes**.

2. Enter the following command to propagate the change throughout the EG:

   ```
   Admin:switch> cryptocfg --commit
   ```

3. Remove the blade from DCX1, slot 4 and plug into DCX2, slot 3.

4. Add the moved blade as a member node to EG2.

Moving an encryption switch from one EG to another in the same fabric

In this example, which is represented in Table 18, you have two EGs, each containing two nodes. You want to move FOS2 from EG1 to EG2.

TABLE 18 Moving a Brocade Encryption Switch from one EG to another EG

Encryption group	Nodes (before move)	Nodes (after move)
EG1	FOS1 (GL) FOS2	FOS1 (GL)
EG2	FOS3 (GL) FOS4	FOS3 (GL) FOS4 FOS2

1. Enter the following command on FOS1 to reclaim the VI/VT WWN base for the Brocade Encryption Switch to be moved out of EG1.

   ```
   Admin:switch> cryptocfg --reclaimWWN -membernode <FOS1_WWN>
   ```

   When prompted, answer **yes**.

2. Enter the following command to propagate the change throughout the EG:

   ```
   Admin:switch> cryptocfg --commit
   ```

3. Enter the following command in FOS1 to eject node FOS2 from the EG:

   ```
   Admin:switch> cryptocfg --eject -membernode <FOS2-WWN>
   ```

4. Enter the following command on FOS1 to deregister the ejected node from the encryption group:

   ```
   Admin:switch> cryptocfg --dereg -membernode <FOS2-WWN>
   ```

5. Enter the following command on FOS2 to clean up the encryption configuration on the deregistered node:

   ```
   Admin:switch> cryptocfg --reclaimWWN -cleanup
   ```

   When prompted, enter **yes** to each prompt.

6. Add FOS2 as a member node to EG2.

Appendix

State and Status Information

A

In this appendix

Encryption engine security processor (SP) states

Table 19 lists the encryption engine security processor (SP) states.

TABLE 19 Encryption engine security processor (SP) states

Encryption engine security processor (SP) state	Description
Not available	SLOT_XXX_SCN has not been received (debug message).
Not Brocade Encryption Switch or DCX	Debug message.
Not Ready	BLADE_RDY_SCN has not been received (debug message).
Fail to connect to blade	Encryption engine is not connected to the blade processor.
Starting	BLADE_RDY_SCN received and initializing.
SPC Connecting to SP	Establishing connection to CP.
SPC Connected to SP	Connection to CP established.
SP Initialized	SP initialization completed.
Waiting for initnode	SP is awaiting initialization. Run **initnode**.
Disabled	SP is disabled.
Encryption engine Faulty	Encryption engine is faulty (BP fault or SP fault). Issue **reboot**.
Waiting for initEE	SP is awaiting initialization. Run **initEE**.
Waiting for regEE	SP has its own certificates but is awaiting FIPS certificates. Run **regEE**.
Waiting for enableEE	Awaiting the explicit enabling of encryption engine. Run **enableEE**.
Operational; Not Online	Encryption engine is operational but offline.
Operational; Need Valid KEK	No current master key or primary or secondary link key. Check the KEK status for more details.
Operational; Need Encryption Group	Encryption engine is operational, but EG is not configured or EG information is not available. Check EG status.
Online	Encryption engine is online.
Zeroized	Encryption engine is zeroized
INVALID	Encryption engine is invalid

Security processor KEK status

Table 20 lists security processor KEK status information.

TABLE 20 Security processor KEK status

KEK type	KEK status[1]	Description
Primary KEK (current MK or primary KV link key)	None	Primary KEK is not configured.
	Mismatch	Primary KEK mismatch between the CP and the SP.
	Match/Valid	Primary KEK at CP matches the one in the SP and is valid.
Secondary KEK (alternate MK or secondary KV link key)	None	Secondary KEK is not configured.
	Mismatch	Secondary KEK mismatch between the CP and the SP.
	Match/Valid	Secondary KEK at CP matches the one in the SP and is valid.
Group KEK	None	Group KEK is not configured.
	Mismatch	Group KEK mismatch between the CP and the SP.
	Match/Valid	Group KEK at the CP matches the one in the SP and is valid.

[1] Only valid in the "encryption engine awaiting encryption group" state and the "encryption engine online" state.

Encrypted LUN states

Table 21 lists encrypted LUN states. Table 22 lists LUN states that are specific to tape LUNs.

TABLE 21 Encrypted LUN states

LUN state	String displayed
UNKNOWN	Unknown
LUN_STATE_UNAVAILABLE	LUN state unavailable.
LUN_STATE_INIT	Initialize
LUN_DISC_START	LUN discovery in progress.
LUN_DISC_COMPLETE	LUN discovery complete.
LUN_SETUP_START	LUN setup
LUN_CLEAR_TEXT	cleartext encryption enabled.
LUN_ENCRYPT	Encryption enabled.
LUN_READONLY_1	Read only (found native metadata while LUN is in DF mode).
LUN_READONLY_2	Read only (found DF metadata while LUN is in native mode).
LUN_READONLY_3	Read only (metadata key is in read-only state).
LUN_WR_META_IN_PROG	Write metadata is in progress.

TABLE 21 Encrypted LUN states (Continued)

LUN_1ST_TIME_REKEY_IN_PROG	First time rekey is in progress.
LUN_KEY_EXPR_REKEY_IN_PROG	Key expired rekey is in progress.
LUN_MANUAL_REKEY_IN_PROG	Manual rekey is in progress.
LUN_DECRYPT_IN_PROG	Data decryption is in progress.
LUN_WR_META_PENDING	Write metadata is pending.
LUN_1ST_TIME_REKEY_PENDING	First time rekey is pending.
LUN_KEY_EXPR_REKEY_PENDING	Key expired rekey is pending.
LUN_MANUAL_REKEY_PENDING	Manual rekey is pending.
LUN_DECRYPT_PENDING	Data decryption is pending.
LUN_LOGIN_REQ	Login in progress.
LUN_LOGIN_BUSY	Login busy.
LUN_LOGIN_TIMEOUT	Login timeout.
LUN_ACCESS_DENIED	Login failure.
LUN_TGT_OFFLINE	Target offline.
LUN_ACCESS_CHK	Not ready (Read/Write access verification in progress).
LUN_DISCOVERY_FAILURE	LUN discovery failure.
LUN_DIS_DEK_GET_API_ERR	Disabled (Get key record API returns error).
LUN_DIS_DEK_GET_CB_ERR	Disabled (Key retrieval from vault failed).
LUN_DIS_DEK_INJECT_API_ERR	Disabled (Inject key API returns error).
LUN_DIS_DEK_INJECT_CB_ERR	Disabled (key injection failure).
LUN_DIS_BAD_KEY_STATE_1	Disabled (New key is in rekey state but encrypt exist data is off).
LUN_DIS_META_KEY_NOT_FOUND	Disabled (unable to retrieve key by key ID found from metadata).
LUN_DIS_META_KEY_MISMATCH_1	Disabled (Meta key is in rekey state but it is not the newest key).
LUN_DIS_META_KEY_MISMATCH_2	Disabled (Meta key does not match with one key found by LUN SN).
LUN_DIS_META_KEY_MISMATCH_3	Disabled (Meta key does not match with any key found by LUN SN).
LUN_DIS_BAD_META_KEY_STATE_1	Disabled (Meta key is old key and in rd/wr but new key is not in rekey).
LUN_DIS_NO_CFG_KEY_ID	Disabled (Data state is encrypted but no key ID provided and metadata does not exist).
LUN_DIS_CREATE_KEY_API_ERR	Disabled (Create new key API returns error).
LUN_DIS_CREATE_KEY_CB_ERROR	Disabled (Create new key failure).
LUN_DIS_ADD_KEY_API_ERR	Disabled (Add new key API returns error).
LUN_DIS_ADD_KEY_CB_ERR	Disabled (Add new key failure).
LUN_DIS_REKEY_ACK_ERR	Disabled (Rekey back with failure).
LUN_DIS_REKEY_DONE_ERR	Disabled (Rekey done with failure).
LUN_DIS_WR_META_ACK_ERR	Disabled (Write metadata back with failure).

TABLE 21 Encrypted LUN states (Continued)

LUN_DIS_WR_META_DONE_ERR	Disabled (Write metadata done with failure).
LUN_DIS_LUN_REMOVED	Disabled (LUN re-discovery detects LUN is removed).
LUN_DIS_LSN_MISMATCH	Disabled (LUN re-discovery detects new device ID).
LUN_DIS_DUP_LSN	Disabled (Duplicate LUN SN found).
LUN_DIS_DISCOVERY_FAIL	Disabled (LUN discovery failure).
LUN_DIS_NO_LICENSE	Disabled (Third party license is required).
LUN_DIS_WRONG_DEV_TYPE	Disabled (Wrong device type found).
LUN_DIS_NOT_SUPPORTED	Disabled (LUN not connected or supported).
LUN_DIS_CFG_KEY_NOT_FOUND	Disabled (Unable to retrieve key by key ID specified from configuration).
LUN_DIS_META_FOUND	Disabled (Data state is cleartext but metadata exists on the LUN).
LUN_DIS_BAD_KEY_STATE_2	Disabled (Data state is encrypted but there is one key which is in rekey state).
LUN_DIS_BAD_KEY_STATE_3	Disabled (Key is in invalid rekey state for encrypted data).
LUN_DIS_BAD_KEY_STATE_4	Disabled (Key is in invalid rekey state while there is one key).
LUN_DIS_BAD_KEY_STATE_5	Disabled (Key is in unknown rekey state).
LUN_DIS_NO_LICENSE_2	Disabled (Found DF metadata while LUN is in native mode and third party license is disabled).
LUN_DIS_LUN_NOT_FOUND	Disabled (LUN not found).
LUN_DIS_GET_DEV_TYPE	Disabled (Inquiry fails).
LUN_DIS_GET_DEV_ID	Disabled (Inquiry device ID page fails).
LUN_DIS_META_FOUND_2	Disabled (Found metadata while LUN is cleartext).
LUN_STATE_UNKNOWN	State of the LUN is unknown.

TABLE 22 Tape LUN states

Internal Names	Console String	Explanation
LUN_DIS_LUN_NOT_FOUND	Disabled (LUN not found)	No logical unit structure in tape module. This is an internal software error. If it occurs, contact Brocade support.
LUN_TGT_OFFLINE	Target Offline	Target port is not currently in the fabric. Check connections and L2 port state.
LUN_DIS_NOT_SUPPORTED	Disabled (LUN not connected or supported)	The target port is active, but this particular Logical Unit is not supported by that target. This indicates a user configuration error.
LUN_DIS_WRONG_DEV_TYPE	Disabled (Wrong device type found)	The logical unit on target port is active, but it is neither a tape or a medium changer. This indicates a user configuration error.
LUN_MEDIUM_CHANGER_ACTIVE	Tape medium changer active	The logical unit is a medium changer, fully ready to handle tapes.
LUN_VALIDATION_PENDING	Tape validation pending	The logical unit is either a tape drive or an attached medium changer, where changer and tape are on same LUN. Since the last LOAD, REWIND, or UNIT ATTENION, no host has attempted to read or write to a tape in this logical unit. There is no way of knowing if a tape is still present, or the encryption state of the tape. A host can issue a READ or WRITE to the logical unit. At that point, it can be determined whether or not a tape is present or needs to be mounted, and whether or not data is ciphertext (encrypted) or cleartext.
LUN_VALIDATION_IN_PROGRESS	Tape validation in progress	The tape module has received the READ or WRITE command that triggers the validation of tape encryption mode, and is in the process of figuring out if the mounted tape medium is encrypted or not. This state should only appear briefly.
LUN_CLEAR_TEXT	Clear text	The tape medium is present, and is in clear text. The encryption switch or blade has full read/write access, because its current tape policy for the medium is also clear text.

TABLE 22 Tape LUN states

LUN_ENCRYPT	Encryption enabled	The tape medium is present, and is in ciphertext (encrypted). The encryption switch or blade has full read/write access, because its current tape policy for the medium is also encrypted. See the Encryption Format field to find out if tape is encrypted in native mode or DataFort-compatible mode.`
LUN_DIS_NO_LICENSE	Disabled (Third party license is required)	The tape medium or its current tape policy is DataFort-compatible mode, but The encryption switch or blade does not have the appropriate license to enable this feature. The tape medium is neither readable nor writable.
LUN_CFG_MISMATCH_CLEARTEXT	Read only (Cleartext tape, policy mismatch	The tape medium is clear text, but current tape policy is not. Mixed modes are not allowed, so the medium is only readable. Attempts to write result in a RASLOG and ABORTED COMMAND returned to host.
LUN_CFG_MISMATCH_COMPATIBLE	Read only (DF_compatible tape, policy mismatch)	The tape medium is encrypted and DataFort-compatible, but the current tape policy is not. Mixed modes are not allowed, so the medium is only readable. Attempts to write result in a RASLOG and ABORTED COMMAND returned to host.
LUN_CFG_MISMATCH_NATIVE	Read only (Native encrypted tape, policy mismatch))	The tape medium is encrypted and native-mode, but the current tape policy is not. Mixed modes are not allowed, so the medium is only readable. Attempts to write result in a RASLOG and ABORTED COMMAND returned to host.

Index